Center for Electronic Imaging Systems

Halftoning and Image Processing Algorithms

Final Progress Report

1 June 1996 - 30 November 1998

DAAH04-96-G-0232
P-35621-PH

U.S. Army Research Office
ATTN: AMXRO-ICA
4300 S. Miami Boulevard
P. O. Box 12211
Research Triangle Park, NC 27709-2211

February 1999

David M. Berfanger

Nicholas George, Principal Investigator
The Institute of Optics
University of Rochester
Rochester, NY 14627

MASTER COPY KEEP FOR REPRODUCTION PURPOSES

REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE February 1999	3. REPORT TYPE AND DATES COVERED Final Progress Report, 6/1/96 - 11/30/98

4. TITLE AND SUBTITLE

Halftoning and Image Processing Algorithms

5. FUNDING NUMBERS

DAAH04-96-G-0232

6. AUTHOR(S)/PI

David M. Berfanger
Nicholas George, Wilson Professor of Electronic Imaging

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES)

The Institute of Optics
University of Rochester
Rochester, NY 14627

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park,, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

ARo 35621. 1-PH

11. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12 b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The focus of this study was theoretical and experimental research on topics in the fields of color halftoning, image processing and compression, and image quality. Our goals in this research were to advance the understanding in image science for our new halftone algorithm and to contribute to image retrieval and noise theory for such imagery. In the field of color halftone printing, research was conducted on deriving a theoretical model of our new halftone algorithm based on a novel resampling of the output pixels, developing halftone algorithms for combining the speed advantages of halftone screening techniques with the quality advantages of error diffusion in the halftoning of color maps, and on color image enhancement for halftone printing. In conjunction with this work, a software development effort was conducted both to implement efficiently the halftoning algorithm itself and to ease its use through a graphical user interface. Research efforts were also conducted in the areas of remote sensing and image compression of color and monochrome images. In image compression we studied the use of controlled blurring to improve both lossless and lossy methods, like DCT-based algorithms. In remote sensing we studied topics in image classification of a scene according to such categories as terrain, vegitation, and image quality.

14. SUBJECT TERMS

Multicolor halftoning; image processing; halftoning algorithms

15. NUMBER IF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL

Enclosure 1

CENTER FOR ELECTRONIC IMAGING SYSTEMS

HALFTONING AND IMAGE PROCESSING ALGORITHMS

FINAL PROGRESS REPORT

1 JUNE 1996 - 30 NOVEMBER 1998

DAAH04-96-G-0232
P-35621-PH

U.S. ARMY RESEARCH OFFICE
ATTN: AMXRO-ICA
4300 S. MIAMI BOULEVARD
P. O. BOX 12211
RESEARCH TRIANGLE PARK, NC 27709-2211

FEBRUARY 1999

DAVID M. BERFANGER

NICHOLAS GEORGE, PRINCIPAL INVESTIGATOR
THE INSTITUTE OF OPTICS
UNIVERSITY OF ROCHESTER
ROCHESTER, NY 14627

Halftoning and Image Processing Algorithms

Final Progress Report

Table of Contents

HALFTONING AND IMAGE PROCESSING ALGORITHMS

SECTION 1: ABSTRACT

The focus of this study was theoretical and experimental research on topics in the fields of color halftoning, image processing and compression, and image quality. Our goals in this research were to advance the understanding in image science for our new halftone algorithm and to contribute to image retrieval and noise theory for such imagery. In the field of color halftone printing, research was conducted on deriving a theoretical model of our new halftone algorithm based on a novel resampling of the output pixels, developing halftone algorithms for combining the speed advantages of halftone screening techniques with the quality advantages of error diffusion in the halftoning of color maps, and on color image enhancement for halftone printing. In conjunction with this work, a software development effort was conducted both to implement efficiently the halftoning algorithm itself and to ease its use through a graphical user interface.

Research efforts were also conducted in the areas of remote sensing and image compression of color and monochrome images. In image compression we studied the use of controlled blurring to improve both lossless and lossy methods, like DCT-based algorithms. In remote sensing we studied topics in image classification of a scene according to such categories as terrain, vegitation, and image quality.

This program of research was heavily leveraged by participation for the ARO-URI in Optoelectronic Systems, the Center in Electronic Imaging Systems (CEIS), and Industrial sponsors including several U.S. corporations who are leaders in the imaging field.

Halftoning and Image Processing Algorithms

Section 2: Scientific Progress and Accomplishments

2.A Statement of the Problem

We report the results of a 3-year study of theoretical and experimental research on topics in color halftoning, image processing, image compression, and image quality. Our goals in this research were to advance the understanding in image science for our new halftone algorithm and to make contributions to image retrieval and noise theory for such imagery. In the field of color halftone printing, we have been conducting and continue to conduct research in connection to the theoretical imaging model of our patented halftone algorithm based on a novel resampling of the output pixels. This includes developing new algorithms to combine the speed advantages of halftone screening techniques with the quality of error diffusion in the halftoning of color maps.

Supporting research was also conducted in color image fidelity, enhancement, and processing for halftone printing. This includes a concentrated effort in the development of image compression algorithms, as well as a more general study of color rendering. In this research we aim to increase both the efficiency and the quality of print produced by our system. Specific areas of activity include the following: improving existing lossy compression algorithms, like DCT-based algorithms; incorporating image classification of a scene according to such categories as terrain recognition and image quality; and studying the longevity of prints produced by our system. This program of research is heavily leveraged by participation from the ARO-URI in Optoelectronic Systems, the Center for Electronic Imaging Systems, and industrial sponsors including several U.S. corporations who are leaders in the imaging field.

2.B Summary of Important Results

With a central objective of providing basic research critical to the mission of the U.S. Army Topographic Engineering Center (TEC), the first year of this project focused on the development of software to operate our new halftone algorithm as part of the QRMP system. In this activity, we were highly successful in developing software in close cooperation with TEC and with the prime contractor, Martin-Marietta. In addition, during the course of this first year's work, three patent applications were filed.

During the second year of this project, under reduced funding, we continued to provide extensive support to refine the system. We completed and delivered three releases of our software as part of our collaborative development of the halftoning module of the QRMP system. Version 3.0 of our software standardized the existing user interface to facilitate operation of the software and enhanced the execution of the expanded functionality of Version 2.0. Features included a multi-process execution model, a straight forward calibration procedure, facilities to save calibration tables for commonly used media and inks, several user controls to adjust the free parameters of the algorithm, the ability to work with several image file formats, and the ability to save individual color channels into separate files. All of this work was accompanied by a vigorous concentration on complete documentation of both the theory and the operation of the software.

During the third year, which proceeded under a no-cost extension of the second year funding, a final version of the software was completed: Htcontrol 3.1. This version of the software corrected reported bugs and included underlying code documentation, as well as a user's manual and resource documentation (see Appendix).

Throughout this study in our theoretical and experimental research, we studied general topics in image science directly applicable to both multicolor printing and image processing. This included further study of our work in controlled blurring and automatic object recognition. We have reported excellent results in these areas with direct applications in secure image transmission, image compression, and image quality assessment. Separately, we studied general areas of color science. We pursued this work with the goal of expanding our ability to accurately and to efficiently render color maps. All of these activities were conducted according to plan as directed through consultation with both TEC and Martin-Marietta.

SECTION 3: PUBLICATIONS AND TECHNICAL REPORTS

"Object recognition and image coding," Nicholas George, B.J. Stossel, and D.M. Berfanger, invited paper presented at Takayanagi Memorial Session, Asia Display '95, Hamamatsu, Japan, October 1995.

"Multiple point impulse responses: controlled blurring and recovery," Bryan J. Stossel and Nicholas George, Opt. Comm. **121**, 156-165 (1995).

"Lensless electronic imaging," Nicholas George, Opt. Comm. **133**, 22-26 (1997).

"Theory for the propagation of short electromagnetic pulses," Nicholas George and Stojan Radic, Opt. Comm. **139**, 1-6 (1997).

Halftoning and Image Processing Algorithms

Section 4: Inventions

"Image data coding and compression system utilizing controlled blurring," **Nicholas George** and Bryan J. Stossel, U.S. Patent No. 5,453,844, September 26, 1995.

"Halftone correction system," **Shen-Ge Wang**, U.S. Patent No. 5,469,267, November 21, 1995.

“Halftone correction systems,” **Shen-Ge Wang**, U.S. Patent No. 5,854,882, December 29, 1998.

Halftoning and Image Processing Algorithms

Section 5: Participating Scientific Personnel

Nicholas George, Principal Investigator; Director, NSF-NYSSTF S/IUCRC Center for Electronic Imaging Systems; Director, ARO-URI Center for Opto-Electronic Systems Research; Wilson Professor of Electronic Imaging; Professor of Optics; and Professor of Electrical Engineering

Shen-ge Wang, Scientist in Optics
Dr. Wang was a major contributor to the TEC project during the first contract period.

(Dr. Wang is currently a Principal Scientist at Xerox Corporation, Webster, NY)

Bryan J. Stossel, Scientist in Optics
Dr. Stossel received his Ph.D. in 1994. Thesis title: "Image processing, coding, and compression with multiple-point impulse response functions."

(Dr. Stossel is currently a scientist at Eastman Kodak Company, Research Laboratories, Rochester, NY)

David M. Berfanger, Ph.D. Fellow
Ph.D. thesis topc: "Automatic pattern recognition using an all digital ring-wedge detector."

William Wade Cook, Ph.D. Fellow.
Ph.D. thesis topic: holographic contouring of large objects

Damon Diehl, Ph.D. Fellow.
Ph.D. hesis topic: holographic optical switch

Wendell Allen Neff, Ph.D. Fellow
Ph.D. thesis topic: image recovery

Jun Ren, M.S. Fellow.
Ms. Ren received her M.S. in April 1997. Master's title: "Atomic force microscopy."

(Ms. Ren is currently pursuing a Ph.D. in California)

Gregory S. Kaufman, Technical Computer Assistant

Appendix: “Using Htcontrol 3.1”

THE CENTER FOR ADVANCED TECHNOLOGY
ELECTRONIC IMAGING SYSTEMS

USING HTCONTROL 3.1

A GRAPHICAL USER INTERFACE FOR THE UNIVERSITY OF ROCHESTER'S HALFTONING ALGORITHM

DOCUMENTATION FOR
HALFTONING AND IMAGE PROCESSING ALGORITHMS

U.S. ARMY RESEARCH OFFICE
ATTN: AMXRO-IC
4300 S. MIAMI BOULEVARD
P.O. BOX 12211
RESEARCH TRIANGLE PARK, NC 27709-2211

DAVID M. BERFANGER

NICHOLAS GEORGE, PRINCIPAL INVESTIGATOR
THE INSTITUTE OF OPTICS
UNIVERSITY OF ROCHESTER
ROCHESTER, NY 14627

ABOUT HTCONTROL

VERSION 3.1

WHAT IS HTCONTROL?

The University of Rochester has developed a revolutionary algorithmic improvement in error-diffusion halftone printing (three patents applied for). Htcontrol is a graphical user interface designed to facilitate the use of this halftoning algorithm. It is the result of an ongoing program of research coordinated with the Army Research Office and the U.S. Army Topographic Engineering Command (TEC), which has included both theoretical and experimental studies in the fields of color halftoning, image processing, image compression, and image quality. The objective of this research has been to provide basic research critical to the mission of the TEC with specific research goals centering on making significant contributions to imaging science in the following areas of interest:

1) Color and monochrome halftone printing systems,

2) Retrieval and noise theory for halftoned imagery,

3) Image compression of color and monochrome scenes, and

4) Automatic object recognition

With Htcontrol, an operator can produce excellent quality halftoned prints with relatively little expert knowledge. At near optimal resolution and with an enhanced color matching ability, the software is especially well suited for producing excellent quality, highly legible multicolor maps.

Conforming to OSF Motif standards, Htcontrol is an intuitive interface that lets the operator customize the output to the software to the special requirements of individual images. Additionally, the software provides features for calibrating its output to different papers and inks, and for saving the individual color separations of input images into separate files.

Htcontrol is a product of the University of Rochester with U.S. patent no. 5,469,267 covering its underlying algorithms. The principal investigator is Dr. Nicholas George, Joseph C. Wilson Professor of Electronic Imaging. Dr. Nicholas George is also the founding director of the Center for Electronic Imaging Systems and the ARO-URI Center for Opto-Electronic Systems Research. Also participating has been Dr. Shen-ge Wang, Dr. Bryan J. Stossel, David M. Berfanger, and Greg Kaufman.

This program of research is greatly enhanced by the local infrastructure in the Rochester electronic imaging community. We would also like to acknowledge the importance of the direct interaction with project engineering scientists at the U.S. Army Topographic Engineering Command. This interaction has served to keep our research efforts relevant allowing us to identify important research problems.

WHAT IS HALFTONING?

A continuous tone image can contain several thousand individual colors; however, typical displays and printers may have available only a few colors for rendering the image. Halftoning is the process whereby a continuous tone image is transformed into a discrete tone image that uses only a limited number of colors for display, storage, or printing. The goal of halftoning is to produce an image that resembles, as closely as possible, the original image. There exist many proposed algorithms for obtaining high quality halftoned images, including methods that attempt to incorporate such factors as detailed models for individual rendering processes and the human visual system. A significant problem to address is the ability to produce output that simulates many gray levels while maintaining the linearity of the grayscale tone reproduction curve. In the case of multicolor halftoning, this relates directly to color fidelity; that is, the halftoned output should agree as much as possible with the continuous tone image data based on some prescribed color standard. We have recently demonstrated experimental results using a newly discovered redefinition of the output pixels in the halftoned image [1,2]. The new method is applicable to all halftoning algorithms and produces high quality images with very nearly linear tone reproduction curves. By providing a straight forward means of correcting for dot overlap, this new method has introduced the possibility of operator independent halftone correction, as well as the real-time monitoring and calibration of printers. The halftoning algorithm implemented by Htcontrol includes a Floyd error-diffussion type halftoning algorithm enhanced with this technique.

WHAT'S NEW IN VERSION 3.1?

New with version 3.1 is a previewer capability that allows the operator to estimate the effects of altering the various parameters of the algorithms. These parameters can be adjusted by the operator using the several user controls available within the interface, including color, black/white threshold, and sharpness. This ability to adapt the algorithm to the particular needs of individual images allows the operator to produce a final print with a near optimal appearance.

While not yet providing true "what-you-see-is-what-you-get" performance, this previewer represents a major improvement in the utility of the user interface. Previously the user had to print hard copy of the halftoned image to see the effects of any edits. Since the halftoning and printing process can take in excess of 10 minutes to complete, the addition of the previewer should significantly reduce the time an operator will need to produce an acceptable print.

Less evadent changes with version 3.1 include several bug fixes and some additional internal code alterations, intended to facilitate future additions to the software. With version 3.0 the user interface was completely divorced from the user front end. Now we have focused on separating file format interpretation from the main body of the halftoning algorithm. This continuing effort should allow for the addition of other file formats to the list of those currently supported, like Erdas .map image files.

FUTURE WORK

As eluted to above, a main topic of effort remaining to be completed is the addition of the software's ability to halftone Erdas .map files. Erdas .map files contain image data (product) along with annotations. Eliminating the need to translate these files to alternate formats before halftoning will speed the user and preserve valuable information about the map being halftoned. This represents a much larger effort than was originally predicted, requiring routines to be written for rendering vector information into a raster format before the data can be halftoned.

Work towards halftoning the .map file format has revealed what could represent a major new area of research. Since vector graphics represents a type of information fundamentally different from the raster-type information currently being dealt with, specialized halftoning could be of value in significantly improving the quality of a final print, which includes this type of information. Along similar lines, since the symbols used in map legends represent what is essentially vector-type information, augmenting the current algorithm with automatic recognition routines for segmenting these symbols from the rest of the scene should provide a significant quality improvement for printed maps.

While recent efforts in software improvements have centered around the refinement of the user interface and its functionality, we feel that a shift of effort towards the improvement of the halftoning algorithm itself represents a substantial opportunity for TEC to produce better maps. Additionally, while excellent experimental results have been obtained, a theoretical analysis is necessary in order to fully understand the consquenses of the novel redefinition of output pixels. Important properties to analyze are the spatial-frequency response of the halftone and the accuracy with which details are reproduced. It is possible that with a complete theoretical description of the new halftone procedure further advances in image quality might be realized. Separately, we have been working toward eliminating inefficiencies of the software in memory management and in computational overhead. While these efforts have only a small effect in the current release continued efforts could significantly reduce the time required to halftone an image.

As this project is nearing completion, we would like to emphasize that the University of Rochester is uniquely capable of continuing to improve the quality of the mapping software available to the TEC. While continued refinements of the user front end of the software are possible, this is especially true when considering the refinement of the halftonig process itself. One is hard pressed to find similar capabilities in imaging research enhanced by the local infrastructure of Rochester's imaging community and leveraged by our other research at the ARO-URI Center for Optoelectronic Systems and the Center for Advanced Technology in Electronic Imaging Systems.

ABOUT THIS MANUAL

This manual is part of a continuing effort to document fully the software written to implement and facilitate the use of the University of Rochester's halftoning algorithm. The manual is designed for use by an Htcontrol operator as an aid in the production of high quality halftoned printing.

The manual contains complete instructions to the operator for operating Htcontrol. Illustrations of all of the windows and controls available to the operator within the graphical user interface have been included. These illustrations are grouped by category corresponding to that used in the menu structure of the software. Additionally, we have included a listing of the resources available to the user, which can be altered to customize the look and feel of the software. Separate documentation covering more fully the technical specifications of the algorithm and the accompanying software are currently being written.

Forward References

1. S. Wang, "Halftone Correction System," U.S. Patent Number 5,469,267, November 1995.

2. S. Wang, "Overlapping correction by centering concept in blue-noise halftoning," presented at the Annual Meeting of the Optical Society of America, Dallas, TX (1994).

CHAPTER

1

GETTING STARTED

INSTALLING THE SOFTWARE

UNPACKING

This chapter describes the steps you need to go through install Htcontrol on a target workstation. Htcontrol 3.1 has been written for and tested on both SUN and HP workstations. The most important factor in a successful build is in the identification of the libraries used in the compilation process. These are detailed more fully later in the separate descriptions for the individual components of the release.

The instalation process begins with the unpacking of the release to a reasonable location of the hard drive of the target workstation. For the rest of this manual, we will call this location `$(HT_INSTALL)`. To unpack the release from the provided 4mm tape, use the following commands:

```
➢ cd $(HT_INSTALL)

➢ tar -x htcontrol_3.1
```

During the unpacking of the distribution, the following directory will be created:

```
➢ $(HT_INSTALL)/htcontrol_3.1.
```

This directory will be refered to as `$(HT_HOME)` in this manual. Note that both `$(HT_INSTALL)` and `$(HT_HOME)` are used only for reference in this manual and are not intended to be an environment variable or string literals.

Version 3.1 of the software release consists of three main components: a tiff library, the vista image display package, and the code implementing the halftoning application. Each are built separately as described below. Additionally, the software requires that the Erdas imaging libraries be installed on the target workstation. These libraries are not packaged with the software release.

THE TIFF LIBRARY

The TIFF library included with version 3.1 is identical to that distributed with the last release with the exception of a few bug fixes. This library has been slightly altered from the freely available version of the library. Two makefiles are included with the distribution: `Makefile.Sun` and `Makefile.HP`. These two makefiles have not been significantly altered from those released with version 3.0. A succesful build of the TIFF library will produce a linkable version in the following directory:

```
➢ $(HT_HOME)/lib
```

To build the TIFF library pick the appropriate makefile and initiate the compilation process with the following commands:

```
➢ cd $(HT_HOME)/tiff

➢ cp Makefile.XXX Makefile

➢ make clean

➢ make
```

Several warning messages are reported during the compilation of the library. These messages persist from earlier versions of the software. As some of these warnings have recently signaled bugs in the software, it would be valuable to evaluate separately the merits of each of these warnings, or better, alter the University of Rochester's software to operate using the standard, freely available version of the TIFF library. However, we are currently having no problems with the TIFF library in its present form.

VISTA

Vista is a freely available image processing software package compatible with Motif and X-Windows. For complete details about installing Vista please refer to the various README and INSTALL files that come included with the ditribution. These files can be found in the $(HT_HOME)/vista directory. We have found the process to be very straight forward requiring alteration of only one file: site.def. This file is found in the $(HT_HOME)/vista/config directory. The site.def file provided with this distribution is the one we have used to successfully install Vista on our Sun workstation. site.def.orig is the original version of the file, which comes with the Vista distribution. Here are the alterations that need to be made:

- ➢ `#define BinDir $(HT_HOME)/bin/vista`
- ➢ `#define GenericLibDir $(HT_HOME)/lib`
- ➢ `#define IncRoot $(HT_HOME)/include`
- ➢ `#define MachineLibDir $(HT_HOME)/lib`
- ➢ `#define LintlibDir $(HT_HOME)/lib/lint`
- ➢ `#define ManRoot $(HT_HOME)/man`
- ➢ `#define SourceDir $(HT_HOME)/vista`
- ➢ `#define XappLoadDir $(HT_HOME)/lib/X11/app-defaults`
- ➢ `#define XbitmapDir $(HT_HOME)/include/X11/bitmaps`

Additionally, we had to make the following alterations as appropriate to the target system:

- ➢ `#define XcolorDatabase /usr/openwin/lib/X11/rgb.txt`
- ➢ #define StandardIncludes

```
#define StandardIncludes -I/usr/include  \
          -I/usr/openwin/include -I/usr/dt/include
```

```
#define LinkerLibDirs -L/usr/lib  \
          -L/usr/openwin/lib/X11 -L/usr/dt/lib
```

These defines tell the system where to find standard includes and libraries, including X-Windows and Motif. You should review the other entries of `site.def` to insure that they are appropriate for your system.

Once the `site.def` file has been updated for your system, the build proceeds as follows:

```
cd $(HT_HOME)/vista
```

```
make World
```

```
make install
```

The University of Rochester's Halftoning Software

Once both the TIFF library and the Vista package have been installed the halftoning software may be compiled. Before compilation the `Makefile` in the `$(HT_HOME)/src` directory must be reviewed to insure that the `INCLUDE` and `LIB` macros are set to values appropriate for your system. Commented lines provide values for libraries and compiler options used to build earlier versions of the halftoning software. The makefile is currently set up for our system and works perfectly. A few minor changes to macros should be all that is needed to compile the software on other machines. Once the makefile is in order, the build continues as follows:

```
cd $(HT_HOME)/src
```

```
make
```

This should result in a working version of four executables in the `$(HT_HOME)/bin` directory: htcontrol, halftone, separations, and patches. All of these executable files are needed to successfully operate all of the features of the halftoning software.

CHAPTER

2

EXPLORING HTCONTROL

BASIC OPERATION OF THE SOFTWARE

USER CUSTOMIZATION

This chapter outlines the basic concepts used in Htcontrol. It assumes that the reader has experience with X-Windows and the Motif window server.

The basic purpose of Htcontrol is to facilitate the production of high-quality halftoned prints. The approach to the halftoning process is as a straightforward filtering process. Htcontrol provides a means for the user to specify an input filename and an output filename. Once these inputs have been specified, the user can initiate the halftoning. Additionally, the interface provides several controls for adjusting the parameters of the halftoning algorithm. These controls allow the user to customize the output of the software for optimum printing of a particular image.

Before you start this chapter you should insure that the Htcontrol and all related software has been successfully installed according to the procedure described in Chapter 1, Getting Started.

After this installation is complete, the expected way for a user to access the functionality of the halftoning software is to create a `.htcontrol` directory in his/her home directory and to place in it a symbolic link to the `$(HT_HOME)` directory. This differs from the implementations of version earlier than version 3.0 and is accomplished by the following set of commands:

```
➢ cd ~

➢ mkdir .htcontrol
```

```
➤ cd .htcontrol

➤ ln -s $(HT_HOME) htcontrol_3.1
```

The `~/.htcontrol` directory is the default location to which Htcontrol writes the default settings resources, and to which the user can keep a default palette file (`default.pal`) and a default calibration file (`default.cal`). Furthermore, it provides the default location to which Htcontrol looks for all of the executable files needed to operate the software, i.e. `~/htcontrol/htcontrol_3.1/bin`. While not necessary, it is convenient for a user to include this directory in his/her command path.

A user specific resource file named Htcontrol should also be included in the users home directory. A template copy of this resource file is provided as an example with this distribution. A complete listing of available resources is provided later in this manual.

All other environment variables, specifically the Erdas environment variables, needed to run previous versions of the halftoning software are still needed for version 3.1. For a complete description of these variables please refer to the appropriate third-party manuals.

RUNNING THE SOFTWARE

Before you begin, insure that your local environment and home directory are setup correctly as described above. If the directory for which the Htcontrol executables live has been included in your command path, entering the following command will begin a Htcontrol session:

```
➤ htcontrol
```

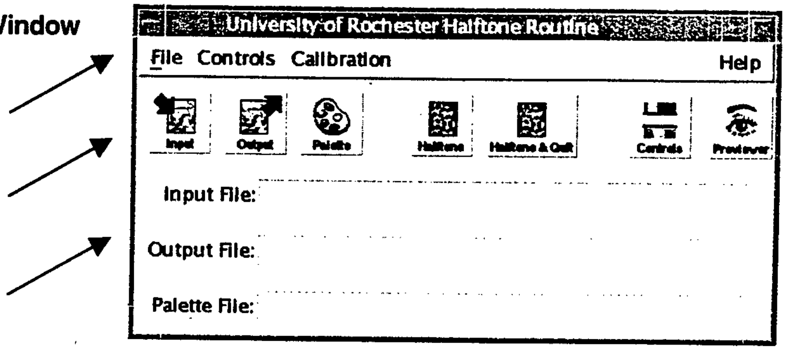
Alternately, you may begin an Htcontrol session by entering the full path of the Htcontrol executable, i.e.,

```
➤ ~/.htcontrol/htcontrol_3.1/bin/htcontrol
```

THE MAIN WINDOW

Upon successful execution of the software, the main window of the application will become visible (See Fig. 1). This window is divided into three regions: a main menu, a toolbar, and a user-input area. As of version 3.0, all other controls for adjusting the parameters of the halftoning algorithm have been relocated from the main window to separate control dialogs.

Fig. 1 The Main Window



Main Menu

Toolbar

Filename Input

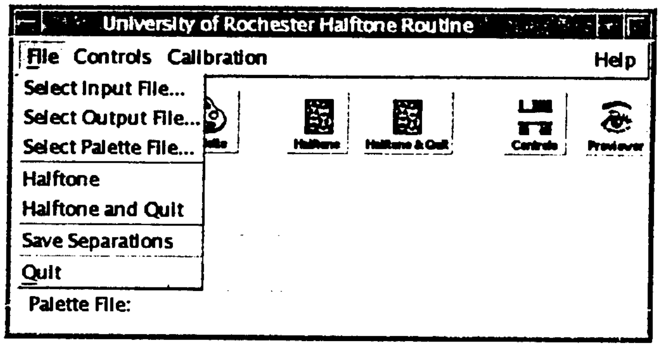
THE MAIN MENU

Using the main menu, a user can initiate all actions needed for successfully producing a halftoned print. This includes selecting the required filenames, requesting dialogs to adjust the parameters of the halftoning algorithm, and initiating the procedure for calibrating the software for a particular printing configuration. Additionally, a simple online help system has been provided.

THE FILE MENU

The File menu allows the operator to specify the necessary files for halftoning: the input file, the output file, and (if required) the palette file. Once these files have been specified the user can then initiate the halftoning process. In doing this the user has the choice of having the software immediately quit after the command has been processed or remain open to initiate additional halftoning jobs. Also, the user can request that the individual color channels of the input image be saved into separate files or that the application be quit without any further processing.

Fig. 2 The File Menu



Select Input File...

This menu option brings forward a standard Motif file selection dialog for selecting the name of the desired input file.

Select Output File...

This menu option brings forward a standard Motif file selection dialog for selecting the name of the desired output file.

Select Palette File...

This menu option brings forward a standard Motif file selection dialog for selecting the name of the desired palette file. This filename is not always required.

Halftone

This menu option initiates the halftoning process using the provided filenames and the current control settings. During the halftoning process a progress dialog is displayed showing the time to completion of the job. This progress dialog provides a Cancel button to abort the job. For a complete description of the available control settings, please refer to Chapter 4, Adjusting the Halftoning Algorithm.

Halftone and Quit

This menu option initiates the halftoning process as above; however, the application is quit after the command has been processed.

Save Separations

This menu option initiates the saving of the individual color channels of the input image into separate files. When this option is selected, a dialog is shown prompting the user for the separation type: either RGB or CMYK. This dialog provides an Ok button to initiate the selected separations job or alternately a Cancel button abort the operation before it is initiated. The filenames used for the separate files are derived from the provided output filename by appending appropriate default filename extensions. During the separations process a progress dialog is displayed showing the time to completion of the job. This progress dialog provides a Cancel button to abort the job before it reaches completion.

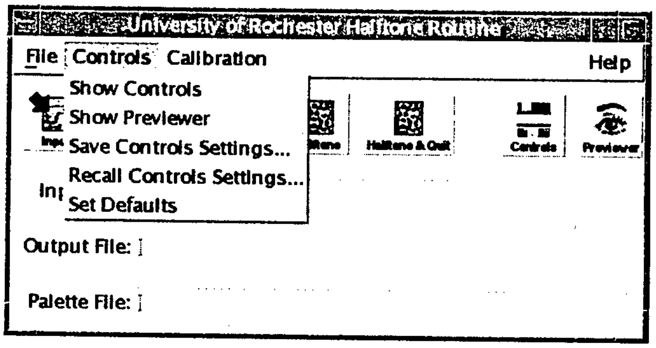
Quit

This menu option terminates application without any further processing. Halftoning and Separations jobs that have already been initiated are not effected by this action.

The Controls Menu

The Controls menu allows the user to request dialogs for adjusting controls setting of the application or to preview the effect of these control settings. Additionally, the user has the options of saving the current control settings to a file for future recall. Finally, an option is provided for the user to set the current controls settings as the default values for future sessions. For a complete description of the available control settings, please refer to Chapter 4, Adjusting the Halftoning Algorithm.

Fig. 3 The Controls Menu



Show Controls

This menu option brings forward a dialog for adjusting the available controls settings of the software. A complete description of this dialog is presented in Chapter 4, Adjusting the Halftoning Algorithm.

Show Previewer

This menu option brings forward a dialog for viewing the selected input file after the current controls settings have been applied. Currently this previewer does not provide true "what-you-see-is-what-you-get" performance; however, it does provide a useful estimation of the effects of these controls settings. A complete description of the operation of the previewer is presented in Chapter 4, Adjusting the Halftoning Algorithm.

Save Controls Settings...

This menu option brings forward a standard Motif file selection dialog for selecting the name of a file in which to save the current controls settings.

Recall Controls Settings...

This menu option brings forward a standard Motif file selection dialog for selecting the name from which to recall previously saved controls settings.

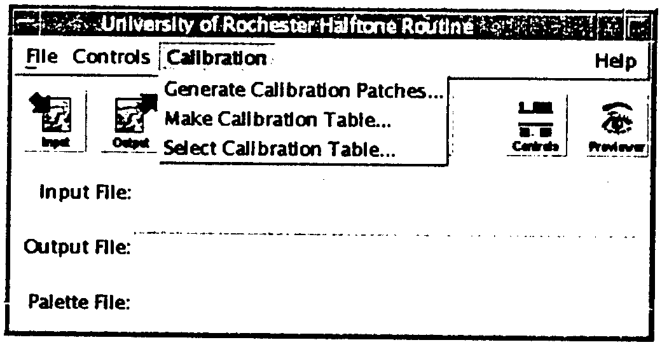
Set Defaults

This menu option tells the software to update its current defaults to the current controls settings. This information is stored in the `.htcontrol` directory assumed to be present in the user's home directory.

THE CALIBRATION MENU

The Calibration menu allows the user to request all functions involved in calibrating of the halftoning algorithm. These include generating calibration patches, calibrating the algorithm, and saving and recalling calibrations tables. For a complete description of the calibration proceedure, please refer to Chapter 3, Calibrating the Halftoning Algorithm.

Fig. 4 The Calibration Menu


University of Rochester Halftone Routine
File
Controls
Calibration
Help
Generate Calibration Patches...
Make Calibration Table...
Select Calibration Table...
Input
Output
Controls
Previewer
Input File:
Output File:
Palette File:

Generate Calibration Patches...

This menu option brings forward a standard Motif file selection dialog for selecting the name of a file in which to save the a printable version of the standard calibration test patches.

Make Calibration Table...

This menu option initiates the calibration procedure, which results in a calibration table describing the dot-overlapping associated with a particular paper and ink combination. A complete description of the calibration procedure is presented in Chapter 3, Calibrating the Halftoning Algorithm.

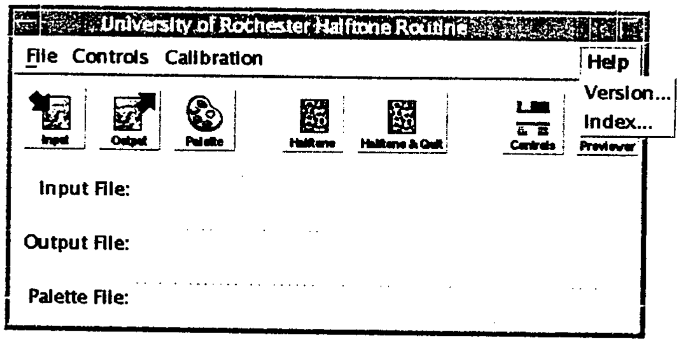
Select Calibration Table...

This menu option brings forward a standard Motif file selection dialog for selecting the name of a file from which to recall a previously generated calibration table.

THE HELP MENU

Htcontrol provides a simple, but useful, online help feature. The help feature can be accessed through the help menu or through the context sensitive help buttons provided throughout the interface.

Fig. 5 The Help Menu



Version...

This menu option brings forward a standard Motif message dialog describing important information about the version of the software, which is currently running.

Index...
This menu option brings forward a dialog displaying a list of topics and associated blurbs, which describe the topic. The topics appear in a selectable list and reflect the content and structure of this manual.

The Toolbar

Htcontrol provides a toolbar to speed the execution of frequently encountered tasks, like requesting file selection dialogs and showing the controls dialog. Each of the buttons in the toolbar have an analogous option in the main menu tree. These include Select Input File, Select Output File, Select Palette File, Halftone, Halftone and Quit, Show Controls, and Show Previewer.

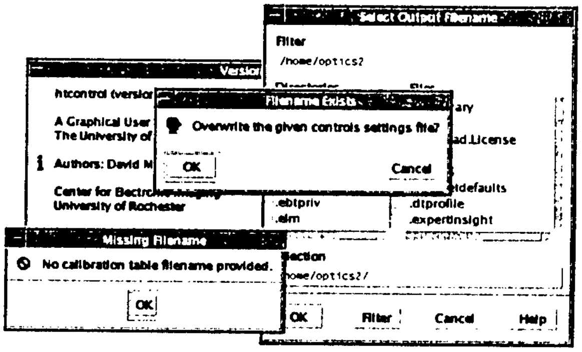
The Filename Entry Area

Htcontrol provides text fields in which the filename required for execution of the halftoning algorithm can be typed directly by the user. These include and input file, an output file, and (if required) a palette file. These filenames can also be selected by requesting an appropriate file selection dialog using either the file menu or the toolbar.

Motif Dialogs

Htcontrol relies heavily on standard Motif dialogs in the implementation of its user interface, including file name selection dialogs, message dialogs, and error dialogs. As the use of these dialogs is highly standardized, we will not explain their use in this manual. For those not familiar with Motif, we refer you to the many texts available on the subject.

Fig. 6 Htcontrol relies heavily on standard Motif dialogs



htcontrol (versio
A Graphical User
The University of
Authors: David M
Center for Electr
University of Rochester

Overwrite the given controls settings file?

No calibration table filename provided.

Filter: /home/optics2

Selection: /home/optics2/

CHAPTER

3

CALIBRATING THE HALFTONING ALGORITHM

THE HEART OF THE SOFTWARE

THE BASIC CONCEPT

Traditional halftoning algorithms are based on an ideal image model consisting of square, non-overlapping pixels. In practice, however, color printers produce small dots at each of the pixel locations. These dots are circular in nature, and they can overlap significantly with dots at adjacent pixel locations. The University of Rochester's halftoning algorithm is based on a patented centering concept for the output pixels. Using this concept, the overlapping between adjacent pixel locations can be characterized for a particular printer, paper, and ink combination. This characterization is accomplished by making a relatively small number of measurements from a representative set of color patches, which were printed using the system for which the algorithm is being calibrated. The heart of our halftoning algorithm is the realization that once the dot-overlapping is characterized for a particular system, any standard halftoning algorithm can be altered to account for this deviation from the ideal image model. The current release of Htcontrol uses a Floyd-type error-diffusion algorithm that incorporates this calibration concept.

OVERVIEW OF THE CALIBRATION PROCEEDURE

As described above, the purpose of the calibration procedure is to optimize the halftoning algorithm for each printer, paper, and ink combination. This proceedure provides a means for attaining accurate color reproduction in the halftoned ouput. The calibration procedure consists of four (4) main steps:

1. Generate the calibration patches and print them using the particular printer, paper, and ink combination,

2. Scan the printed calibration patches into the workstation,

3. Make the calibration table using the scanned input file, and

4. Select the resulting data as the current calibration table for the software.

To complete the printing and scanning of the test patches in this proceedure, it is necessary to use some third-party software. Htcontrol provides the functionality necessary to complete the remaining steps. This functionality can be accessed through the "Calibration" menu, described in Chapter 2, Exploring Htcontrol.

GENERATING THE CALIBRATION PATCHES

Upon selecting the "Generate Calibration Patches..." menu item from the "Calibration" menu, a standard Motif file selection dialog is presented to the operator. The dialog allows for the selection of a name for the file in which to store a printable version of the standardized page containing the calibration patches. If an existing filename is chosen the operator is prompted to verify permission to overwrite the old file. The output file will be an HPGL-2 file containing all of the data needed to render the calibration patches. To print this file, the operator will need to employ additional third-party software. Subsequently, this resulting print must be scanned into the workstation, which also will require additional third-party software.

MAKING THE CALIBRATION TABLE

Calibrating the University of Rochester's halftoning algorithm centers around generating an overlap correction table. This table is used to correct for differing effects of various printer-paper-ink combinations. Calibration tables are generated from scanned images of a predefined set of test patches, as described above.

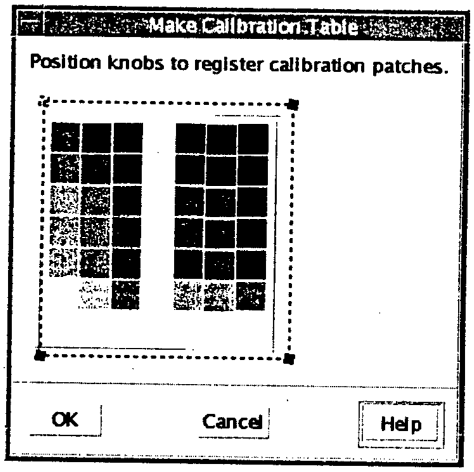
Required parameters for generating a calibration table include an input filename of a file containing a scanned image of the test patches and an output filename designating where to save the generated calibration table data. Allowed file formats for the input file are TIFF and Tangent image files. For Tangent image files a Palette filename is also required. These parameters must be entered into the corresponding text fields of the main window before the calibration table can be generated. For additional information of entering these file names please refer to Chapter 2, Exploring Htcontrol.

The generation of the calibration table is initiated by selecting the "Make Calibration Table..." menu item in the "Calibration" menu. Upon selecting this menu item a dialog is shown requesting the coordinates of the bounding box surrounding the test patches (See Fig. 7). These coordinates are entered using the mouse. The left mouse button can be used to select one of the four (4) knobs of the selection rectangle by clicking inside the knob. The currently selected knob is drawn as an unfilled square. The other three knobs are drawn as filled squares. Once a knob is selected its position can be changed by moving the mouse pointer to the desired location and clicking the left mouse button. After all four knobs are correctly positioned pressing the Ok button finishes the generation of the calibration table to the output file.

Selecting a Calibration Table

A calibration table is needed by the halftoning algorithm to correct for color variations between differing inks and papers. The operator can select the filename of the file containing the appropriate calibration table by choosing the "Select Calibration Table..." menu item in the "Calibration" menu. Upon selecting this menu item a standard Motif file selection dialog is shown for the operator to enter the desired filename.

Fig. 7 The Make Calibration Table Dialog



Make Calibration Table

Position knobs to register calibration patches.

CHAPTER 4

ADJUSTING THE HALFTONING ALGORITHM

OPTIMIZING THE SOFTWARE FOR INDIVIDUAL IMAGE REQUIREMENTS

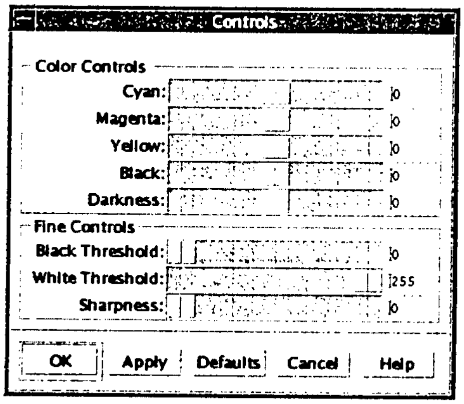
THE CONTROLS DIALOG

Beginning with version 3.0, all of the controls for customizing the appearance of the final halftoned print have been located on a single controls dialog (See Fig. 8). The controls available on this dialog provide the operator with a visual means of adjusting all of the various parameters to the halftoning algorithm. The controls belong to one of two groups: color controls and fine controls.

The vlaue of each of the individual controls on the controls dialog can be changed in on of two ways: a scale can be manipulated using the mouse or a desired value can be entered directly into a text field.

There are five action buttons on the controls dialog: Ok, Apply, Defaults, Cancel, and Help. Pressing the Ok button applies the currently displayed controls settings and dismisses the dialog. Pressing the Apply button applies the currently displayed controls settings without dismissing the dialog. Pressing the Defaults button resets all of the controls to their default value. Pressing the Cancel

Fig. 8 The Controls Dialog



Controls

Color Controls

Cyan: 0

Magenta: 0

Yellow: 0

Black: 0

Darkness: 0

Fine Controls

Black Threshold: 0

White Threshold: 255

Sharpness: 0

button Dismisses the dialog without applying the currently displayed settings. Finally, pressing the Help button calls forward the Help dialog.

THE COLOR CONTROLS

The amount of color correction used in the halftone algorithm can be adjusted by way of the color controls on the controls dialog. These include Cyan, Magenta, Yellow, and Black controls. Additionally, a Darkness control exists which adjusts the other four equally. A more positive value results in more of the respective color in the halftoned image. A more negative value results in less of the respective color in the halftoned output.

THE FINE CONTROLS

Fine controls for the halftoning algorithm include white and black clipping values, as well as a level of sharpness enhancement. These settings are not for general use but may be adjusted if necessary.

The clipping behavior of the halftoning algorithm is as follows. First, the input color values are clipped by the black value and scaled to range from 0 to 255. The resulting data are then clipped by the white threshold value and again scaled. Raising the black clipping value causes more of the darkest colors to be clipped to black. Similarly, lowering the white clipping value causes more of the lightest colors to be rendered as white. These values can be used to reduce background noise in constant areas of the image.

The sharpness parameter, adjusts the behavior of the halftoning algorithm in the neighborhood of high contrast features. Its effect is highly dependent on the image being halftoned.

SAVING THE CONTROLS SETTINGS

Current controls settings can be saved to a file for later use. This is initiated by selecting the "Save Controls Settings..." menu item in the "Controls" menu. Upon selecting this menu item a standard Motif file selection dialog is shown for the operator to select a filename for the file in which to store the controls settings data.

RECALLING THE CONTROLS SETTINGS

Previously saved controls settings can be recalled from a file. This is initiated by selecting the "Recall Controls Settings..." menu item in the "Controls" menu. Upon selecting this menu item a standard Motif file selection dialog is shown for the operator to select the desired file.

SETTING THE DEFAULT CONTROLS SETTINGS

Default controls settings can be set in the user resource file, as well as by Htcontrol itself. Examples resource file entries for each of the controls settings application resources are given below:

- `*theCyanDefaultValue: 0`
- `*theMagentaDefaultValue: 0`
- `*theYellowDefaultValue: 0`
- `*theBlackDefaultValue: 0`
- `*theDarknessDefaultValue: 0`
- `*theWhiteThresholdDefaultValue: 255`
- `*theBlackThresholdDefaultValue: 0`
- `*theSharpnessDefaultValue: 0`

For additional information about resources used by Htcontrol, please refer to Chapter 5, Htcontrol Resources.

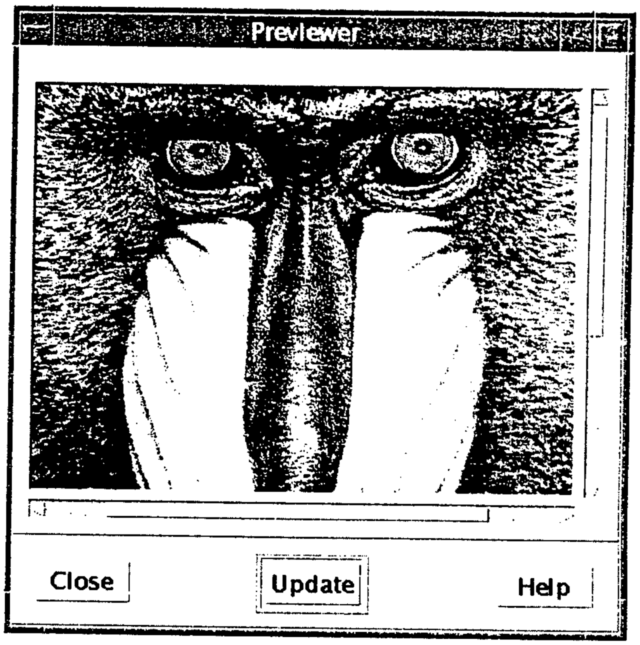
Default values specified in the user resource file can be overridden by values set default settings file. This file can be generated from the current controls settings by selecting the "Set Defaults" menu item in the "Controls" menu. The filename of the controls default settings is defined by the `theControlsDefaultsFilename` resource, which can be customized in a users resource file. The factor setting is as follows:

-
  ```
  *theControlsDefaultsFilename:
  ~/.htcontrol/DefaultSettings
  ```

THE PREVIEWER DIALOG

A previewer dialog is being developed for use with Htcontrol. Version 3.1 of the software includes an initial version of this dialog (See Fig. 9). This preliminary design does not perform at the desired "what-you-see-is-what-you-get" level; however, it does provide some useful insight into the effects of the controls setting on the resulting print.

Fig. 9 The Previewer Dialog



Htcontrol Resource File Documentation

Like most X-windows applications, Htcontrol is designed so that many of its resources can be modified by the user at run-time. As Htcontrol begins execution, it loads data from user-specified files. The data from this file is used to build the *resource database*, containing both information about the configuration of the widgets in the application and the settings of various data attributes which affect operation.

Below, we detail the available configurable resources for Htcontrol(version 3.0). Also included is a brief description of the writing of resource files and the setting of environment variable to allow Htcontrol to find the appropriate resource file. For additional information about the general use of resource files in X-windows we suggest the books listed under References.

1. Htcontrol Resources

A major goal in the development of htcontrol (version 3.0) was to bring its graphical user interface into compliance with OSF/Motif standards. This included the application's handling of user configurable resources. Htcontrol only hard-codes the values of those resources that are essential to program operation. The rest of the resources are left user configurable. Reasonable default values for configurable resources are provided within the executable. Resources for htcontrol fall into three categories: widget resources, application resources, and errror messages. This section contains a brief description of the resource types.

1.1. Widget Resources

A widget is a generic term for a graphical user-interface component. Motif widgets are prolific in their use of resources. For each widget class, there are many resources that provide control over the appearance and functionality of the widgets: color, text fonts, size, etc. For a complete listing of the configurable resources for Motif widget classes see Ref. 1. A full listing of the names of the widgets, their class type, and their location in htcontrol appears in Appendix A.

1.2. Error Messages

1.3. Application Resources

Htcontrol has several user deferable attributes which directly affect the operation of the application: Default setting values, the Calibration filename, the default palette filename, etc.

2. Writing Resource Files

Resource files are ASCII text files containing line entries consisting of either a resource name or a resource class name and a resource value separated by a colon(:). For additional information about resource files see Ref. 2.

2.1. Resource Names

Widgets in htcontrol are created in a heirarchy that starts with a top-level shell and is followed by successive generations of child widgets. To set the resources of a child widget in a resource file the full widget path from the top-level shell to that child widget must be specified. Pathnames are specified as a list of parent widgets starting from the top-level shell, separated by a period and ending with the child widget you want to change. For example, to set the labelString resource for theInputFilenameLabel, the following line in the htcontrol application would be placed in the resource file.

```
    Htcontrol.          .theInputFilenameLabel.LabelString:
Input File
```

Being widget resource themselves Error message resources are set in the same way. With the Error message dialog being a direct child of the TopLevelShell.

2.2 Wildcards

A wildcard (*) can also be used within a resource pathname allowing for both shorter names and the setting of multiple resources with a single line. For example, the following resource command could also be used to set the labelString for theInputFilenameLabel as above:

```
    Htcontrol*theInputFilenameLabel.LabelString:  Input File
```

Similiarly, a user can set the background color resource for every widget in the application using the command:

```
    Htcontrol*background:  blue
```

Within the Htcontrol resource file this could even be further shortened to the following:

```
    *background:  blue
```

Resource commands can also use the widget's class name instead of the widget name. For example, the following line would set all LabelStrings in the **xmLabel** class.

```
   Htcontrol.     .theInputFilenameLabel.xmLabel.LabelString:Input
File
```

If, through the use of wildcards or otherwise, two or more comands within a resource file specify a common resource name several rules of precedence exist to resolve the conflict.

- Later occuring resource commands have precedence.
- More specific resource commands have precedence.
- Resource names have precedence over resource class names.

2.3 Resource File Errors

Resource files are hard to debug since most errors are quietly ignored. Release 5 has included a resource command called StringConversionWarnings which will print out an error message if problems occur when the text from the resource file is converted into actual values. The messages are not always clear, but may help. In order to turn on the warning, add the following the command line to your resource file. (also see example resource file in Appendix E)

```
    *StringConversionWarnings:  on
```

3. Setting Environment Variables

Once the resource file has been written it must be placed so that the program will find its location. There are two locations that the program will look for by default: a user's home directory and /usr/lib/X11/app-defaults. However, in X applications there is a lot of freedom as to the location of application files. Here applications are told where to look through environment variables. Table 1 lists possible locations for resource files in order of precedence. We find the most correct place for application resource files to be is in a user's home directory. This avoids the need for setting user environment variables, which often serve to complicate matters.

Table 1

References

Ref. 1.

The Definitive Guide to the X Window System
Motif Reference Manual for OSF/Motif release 1.2
Volume Six B, Motif Edition
author: Paula M. Ferguson
O'Reilly & Associates Inc. 1993

Ref. 2.

Power Programming... Motif
second edition, version 1.2, revised and expanded
authors: Eric F. Johnson and Kevin Reichard
MIS:Press Books 1993

APPENDIX A: Widget Names and Classes

theMainWindow	-	xmMainWindowWidgetClass
theMenuBar	-	xmRowColumnWidgetClass
theFilePulldownMenu	-	xmRowColumnWidgetClass
theFileCascadeButton	-	xmCascadeButtonWidgetClass
theInputFilenameMenuItem	-	xmPushButtonWidgetClass
theOutputFilenameMenuItem	-	xmPushButtonWidgetClass
thePaletteFilenameMenuItem	-	xmPushButtonWidgetClass
fileMenuItemSeparator1	-	xmSeparatorGadgetClass
theHalftoneMenuItem	-	xmPushButtonWidgetClass
theHalftoneAndQuitMenuItem	-	xmPushButtonWidgetClass
fileMenuItemSeparator2	-	xmSeparatorGadgetClass
theSaveSeparationsMenuItem	-	xmPushButtonWidgetClass
fileMenuItemSeparator3	-	xmSeparatorGadgetClass
theExitMenuItem	-	xmPushButtonWidgetClass
theControlsPulldownMenu	-	xmRowColumnWidgetClass
theControlsCascadeButton	-	xmCascadeButtonWidgetClass
theShowHideControlsMenuItem	-	xmPushButtonWidgetClass
theShowHidePreviewerMenuItem	-	xmPushButtonWidgetClass
theSaveSettingsMenuItem	-	xmPushButtonWidgetClass
theRecallSettingsMenuItem	-	xmPushButtonWidgetClass
theCalibrationPulldownMenu	-	xmRowColumnWidgetClass
theCalibrationCascadeButton	-	xmCascadeButtonWidgetClass
theGeneratePatchesMenuItem	-	xmPushButtonWidgetClass
theMakeTableMenuItem	-	xmPushButtonWidgetClass

theRecallTableMenuItem	-	xmPushButtonWidgetClass
theHelpPulldownMenu	-	xmRowColumnWidgetClass
theHelpCascadeButton	-	xmCascadeButtonWidgetClass
theVersionMenuItem	-	xmPushButtonWidgetClass
theIndexMenuItem	-	xmPushButtonWidgetClass
theWorkAreaForm	-	xmFormWidgetClass
theToolBarForm	-	xmFormWidgetClass
theInputFilenamePushButton	-	xmPushButtonWidgetClass
theOutputFilenamePushButton	-	xmPushButtonWidgetClass
thePaletteFilenamePushButton	-	xmPushButtonWidgetClass
theHalftonePushButtton	-	xmPushButtonWidgetClass
theHalftoneAndQuitPushButton	-	xmPushButtonWidgetClass
theControlsPushButton	-	xmPushButtonWidgetClass
thePreviewerPushButton	-	xmPushButtonWidgetClass
theFilenameEntryForm	-	xmFormWidgetClass
theInputFilenameForm	-	xmFormWidgetClass
theInputFilenameLabel	-	xmLabelWidgetClass
theInputFilenameTextField	-	xmTextFieldWidgetClass
theOutputFilenameForm	-	xmFormWidgetClass
theOutputFilenameLabel	-	xmLabelWidgetClass
theOutputFilenameTextField	-	xmTextFieldWidgetClass
thePaletteFilenameForm	-	xmFormWidgetClass
thePaletteFilenameLabel	-	xmLabelWidgetClass
thePaletteFilenameTextField	-	xmTextFieldWidgetClass
theControlsDialog	-	transientShellWidgetClass
theControlsMessageBox	-	xmMessageBoxWidgetClass
theControlsForm	-	xmFormWidgetClass

theControlsColorFrame	-	xmFrameWidgetClass
theControlsColorForm	-	xmFormWidgetClass
theControlsColorLabel	-	xmLabelWidgetClass
theControlsFineFrame	-	xmFrameWidgetClass
theControlsFineForm	-	xmFormWidgetClass
theControlsFineLabel	-	xmLabelWidgetClass
theControlsApplyButton	-	xmPushButtonWidgetClass
theControlsResetButton	-	xmPushButtonWidgetClass
theControlsCyanForm	-	xmFormWidgetClass
theControlsCyanLabel	-	xmLabelWidgetClass
theControlsCyanScale	-	xmScaleWidgetClass
theControlsCyanTextField	-	xmTextFieldWidgetClass
theControlsMagentaForm	-	xmFormWidgetClass
theControlsMagentaLabel	-	xmLabelWidgetClass
theControlsMagentaScale	-	xmScaleWidgetClass
theControlsMagentaTextField	-	xmTextFieldWidgetClass
theControlsYellowForm	-	xmFormWidgetClass
theControlsYellowLabel	-	xmLabelWidgetClass
theControlsYellowScale	-	xmScaleWidgetClass
theControlsYellowTextField	-	xmTextFieldWidgetClass
theControlsBlackForm	-	xmFormWidgetClass
theControlsBlackLabel	-	xmLabelWidgetClass
theControlsBlackScale	-	xmScaleWidgetClass
theControlsBlackTextField	-	xmTextFieldWidgetClass
theControlsDarknessForm	-	xmFormWidgetClass
theControlsDarknessLabel	-	xmLabelWidgetClass
theControlsDarknessScale	-	xmScaleWidgetClass

theControlsDarknessTextField	-	xmTextFieldWidgetClass
theControlsBlackThresholdForm	-	xmFormWidgetClass
theControlsBlackThresholdLabel	-	xmLabelWidgetClass
theControlsBlackThresholdScale	-	xmScaleWidgetClass
theControlsBlackThresholdTextField	-	xmTextFieldWidgetClass
theControlsWhiteThresholdForm	-	xmFormWidgetClass
theControlsWhiteThresholdLabel	-	xmLabelWidgetClass
theControlsWhiteThresholdScale	-	xmScaleWidgetClass
theControlsWhiteThresholdTextField	-	xmTextFieldWidgetClass
theControlsSharpnessForm	-	xmFormWidgetClass
theControlsSharpnessLabel	-	xmLabelWidgetClass
theControlsSharpnessScale	-	xmScaleWidgetClass
theControlsSharpnessTextField	-	xmTextFieldWidgetClass
theVersionDialog	-	xmMessageBoxWidgetClass
theHelpDialog	-	topLevelShellWidgetClass
theHelpDialogMessageBox	-	xmMessageBoxWidgetClass
theHelpDialogPanedWindow	-	xmPanedWindowWidgetClass
theHelpDialogTopicsForm	-	xmFormWidgetClass
theHelpDialogTopicsLabel	-	xmLabelWidgetClass
theHelpDialogTopicsList	-	xmListWidgetClass
theHelpDialogTextForm	-	xmFormWidgetClass
theHelpDialogTextLabel	-	xmLabelWidgetClass
theHelpDialogScrolledText	-	xmTextedWidgetClass
theExitMenuItem	-	xmPushButtonWidgetClass
theInputFileSelectionDialog	-	xmFileSelectionBoxWidgetClass
theOutputFileSelectionDialog	-	xmFileSelectionBoxWidgetClass
thePaletteFileSelectionDialog	-	xmFileSelectionBoxWidgetClass

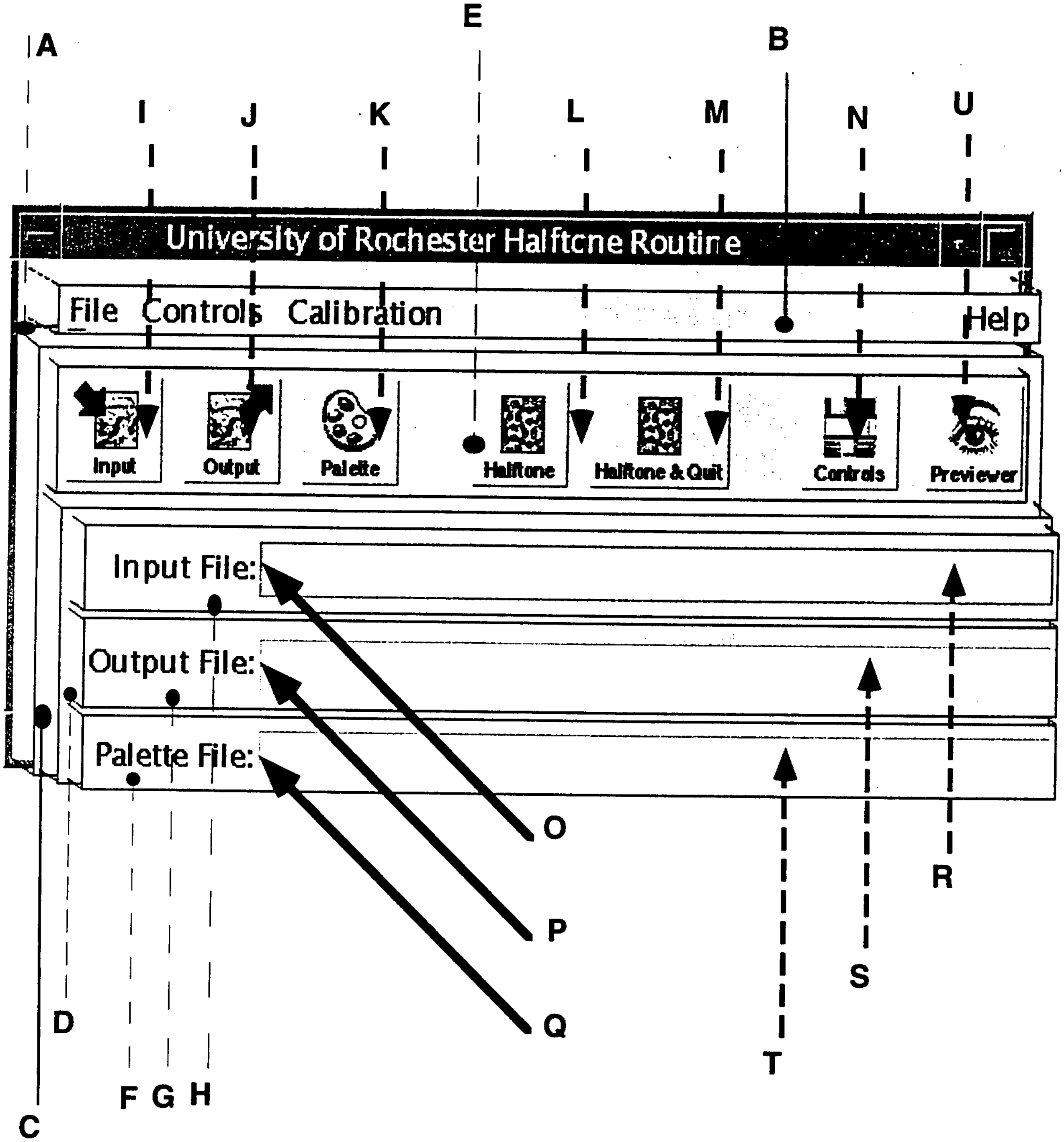
theRecallSettingsFileSelectionDialog	-	xmFileSelectionBoxWidgetClass
theRecallCalibrationFileSelectionDialog	-	xmFileSelectionBoxWidgetClass
theGeneratePatchesFileSelectionDialog	-	xmFileSelectionBoxWidgetClass
theSaveSettingsFileSelectionDialog	-	xmFileSelectionBoxWidgetClass
theHalftoneMessageBox	-	xmMessageBoxWidgetClass
theHalftoneInfoAreaForm	-	xmFormWidgetClass
theHalftoneInputFilenameForm	-	xmFormWidgetClass
theHalftoneInputFilenameLabel	-	xmLabelWidgetClass
theHalftoneInputFilenameTextField	-	xmTextFieldWidgetClass
theHalftoneOutputFilenameForm	-	xmFormWidgetClass
theHalftoneOutputFilenameLabel	-	xmLabelWidgetClass
theHalftoneOutputFilenameTextField	-	xmTextFieldWidgetClass
theProgressScrollBar	-	xmScrolledBarWidgetClass
theSetDefaultSettingMenuItem	-	xmPushButtonWidgetClass
theSeparationTypeSelectionForm	-	xmFormWidgetClass
separationsTypeSelectionDialog	-	xmMessageBoxWidgetClass
theSeparationsTypeSelectionRadioBox	-	xmRowColumnWidgetClass
theRGBSeparationsTypeToggleButton	-	xmToggleButtonWidgetClass
theCMYKSeparationsTypeToggleButton	-	xmToggleButtonWidgetClass
theMultibandSelectionDialog	-	xmMessageBoxWidgetClass
aURMultibandSelectionDialog	-	xmRowColumnWidgetClass
aURSpinBoxForm	-	xmFormWidgetClass
aCalibrationDialogMessageBox	-	xmMessageBoxWidgetClass
aCalibrationDialogForm	-	xmFormWidgetClass
aCalibrationDialogScrolledWindow	-	xmScrolledWindowWidgetClass
aCalibrationDialogForm2	-	xmFormWidgetClass
aCalibrationDialogImageView	-	vImageViewWidgetClass

theSeparationsMessageBox	-	xmMessageBoxWidgetClass
theSeparationsInfoAreaForm	-	xmFormWidgetClass
theSeparationsInputFilenameForm	-	xmFormWidgetClass
theSeparationsInputFilenameLabel	-	xmLabelWidgetClass
theSeparationsInputFilenameTextField	-	xmTextFieldWidgetClass
theRedOutputFilenameForm	-	xmFormWidgetClass
theRedOutputFilenameLabel	-	xmLabelWidgetClass
theRedOutputFilenameTextField	-	xmTextFieldWidgetClass
theGreenOutputFilenameForm	-	xmFormWidgetClass
theGreenOutputFilenameLabel	-	xmLabelWidgetClass
theGreenOutputFilenameTextField	-	xmTextFieldWidgetClass
theBlueOutputFilenameForm	-	xmFormWidgetClass
theBlueOutputFilenameLabel	-	xmLabelWidgetClass
theBlueOutputFilenameTextField	-	xmTextFieldWidgetClass
theCyanOutputFilenameForm	-	xmFormWidgetClass
theCyanOutputFilenameLabel	-	xmLabelWidgetClass
theCyanOutputFilenameTextField	-	xmTextFieldWidgetClass
theMagentaOutputFilenameForm	-	xmFormWidgetClass
theMagentaOutputFilenameLabel	-	xmLabelWidgetClass
theMagentaOutputFilenameTextField	-	xmTextFieldWidgetClass
theYellowOutputFilenameForm	-	xmFormWidgetClass
theYellowOutputFilenameLabel	-	xmLabelWidgetClass
theYellowOutputFilenameTextField	-	xmTextFieldWidgetClass
theBlackOutputFilenameForm	-	xmFormWidgetClass
theBlackOutputFilenameLabel	-	xmLabelWidgetClass
theBlackOutputFilenameTextField	-	xmTextFieldWidgetClass
theCloseMenuItem	-	xmPushButtonWidgetClass

theRedSpinBoxLabel	-	xmLabelWidgetClass
theGreenSpinBoxLabel	-	xmLabelWidgetClass
theBlueSpinBoxLabel	-	xmLabelWidgetClass
theRedSpinBoxTextField	-	xmTextFieldWidgetClass
theGreenSpinBoxTextField	-	xmTextFieldWidgetClass
theBlueSpinBoxTextField	-	xmTextFieldWidgetClass
theRedSpinBoxForm	-	xmFormWidgetClass
theGreenSpinBoxForm	-	xmFormWidgetClass
theBlueSpinBoxDorm	-	xmFormWidgetClass
theRedIncrementArrowButton	-	xmPushButtonWidgetClass
theGreenIncrementArrowButton	-	xmPushButtonWidgetClass
theBlueIncrementArrowButton	-	xmPushButtonWidgetClass
theRedDecrementArrowButton	-	xmPushButtonWidgetClass
theGreenDecrementArrowButton	-	xmPushButtonWidgetClass
theBlueDecrementArrowButton	-	xmPushButtonWidgetClass
theMultibandSelectionDialog_geometryManagerForm	-	xmFormWidgetClass

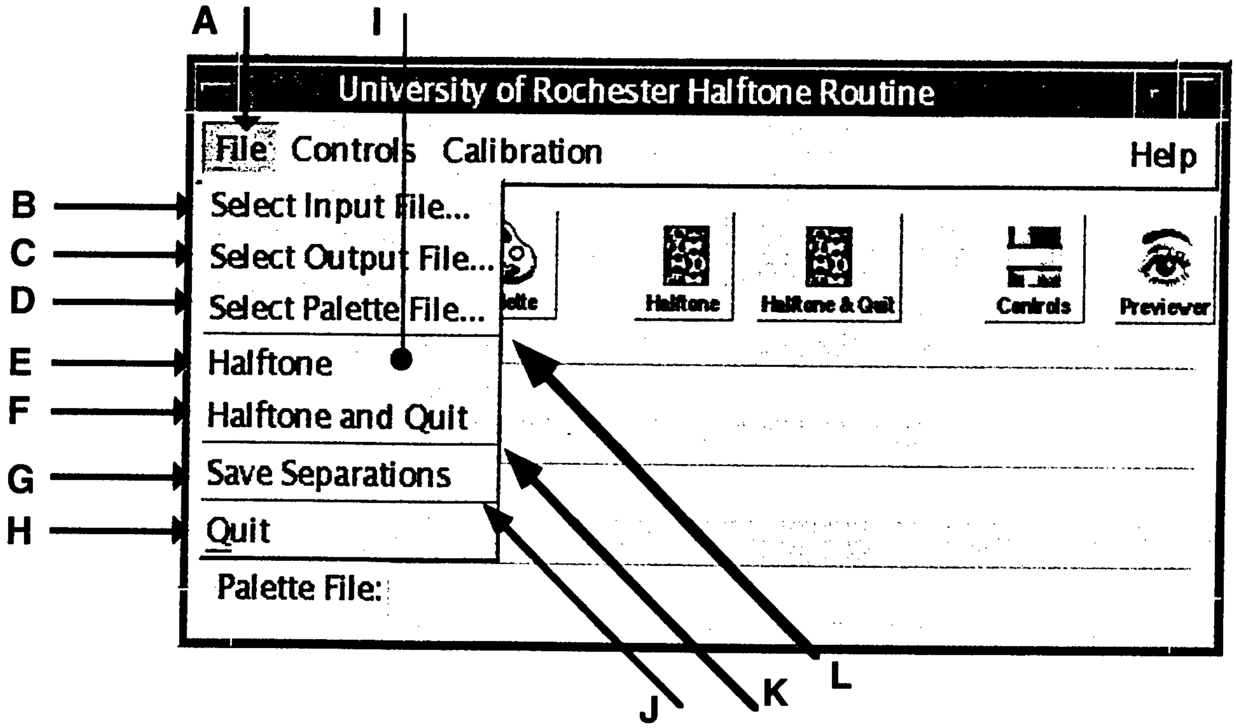
Appendix B: Widget Hierarchy

1. The Main Window



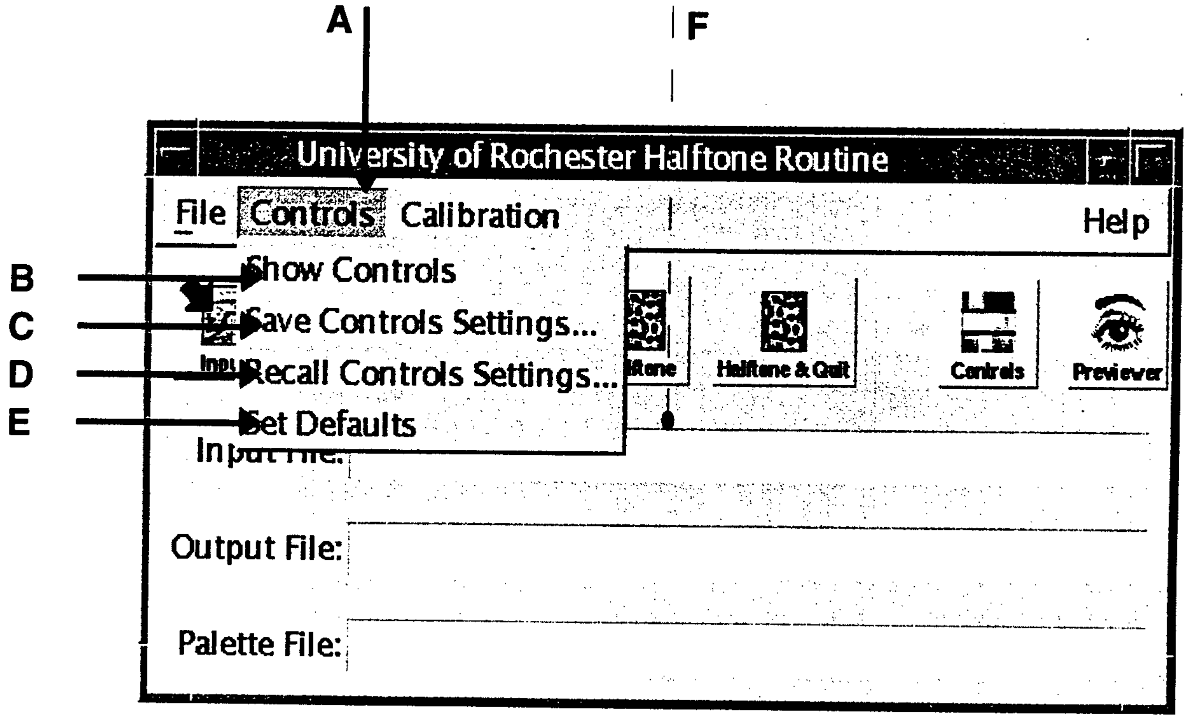
	Widget Names	Widget Classes
A	theMainWindow	xmMainWindowWidgetClass
B	theMenuBar	xmRowColumnWidgetClass
C	theWorkAreaForm	xmFormWidgetClass
D	theFilenameEntryForm	xmFormWidgetClass
E	theToolBarForm	xmFormWidgetClass
F	thePaletteFilenameForm	xmFormWidgetClass
G	theOutputFilenameForm	xmFormWidgetClass
H	theInputFilenameForm	xmFormWidgetClass
I	theInputFilenamePushButton	xmPushButtonWidgetClass
J	theOutputputFilenamePushButton	xmPushButtonWidgetClass
K	thePaletteFilenamePushButton	xmPushButtonWidgetClass
L	theHalftonePushButton	xmPushButtonWidgetClass
M	theHalftoneAndQuitPushButton	xmPushButtonWidgetClass
N	theControlsPushButton	xmPushButtonWidgetClass
O	theInputFilenameLabel	xmLabelWidgetClass
P	theOutputFilenameLabel	xmLabelWidgetClass
Q	thePaletteFilenameLabel	xmLabelWidgetClass
R	theInputFilenameTextField	xmTextFieldWidgetClass
S	theOutputFilenameTextField	xmTextFieldWidgetClass
T	thePaletteFilenameTextField	xmTextFieldWidgetClass
U	thePaletteFilenamePushButton	xmPushButtonWidgetClass

2. The File Menu



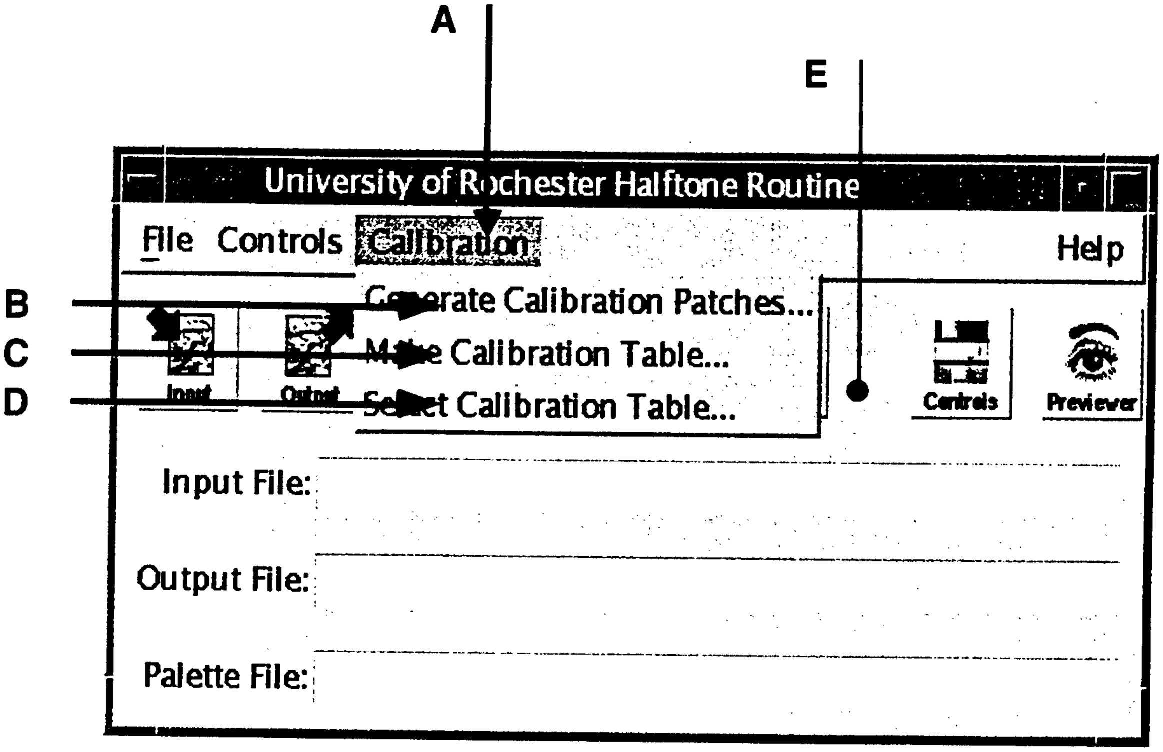
	Widget Names	Widget Classes
A	theFileCascadeButton	xmCascadeButtonWidgetClass
B	theInputFilenameMenuItem	xmPushButtonWidgetClass
C	theOutputFileNameMenuItem	xmPushButtonWidgetClass
D	thePaletteFilenameMenuItem	xmPushButtonWidgetClass
E	theHalftoneMenuItem	xmPushButtonWidgetClass
F	theHalftoneAndQuitMenuItem	xmPushButtonWidgetClass
G	theSaveSeparationsMenuItem	xmPushButtonWidgetClass
H	theExitMenuItem	xmPushButtonWidgetClass
I	theFilePulldownMenu	xmRowColumnWidgetClass
J	fileMenuItemSeparator1	xmPushButtonWidgetClass
K	fileMenuItemSeparator2	xmPushButtonWidgetClass
L	fileMenuItemSeparator3	xmPushButtonWidgetClass

3. The Controls Menu



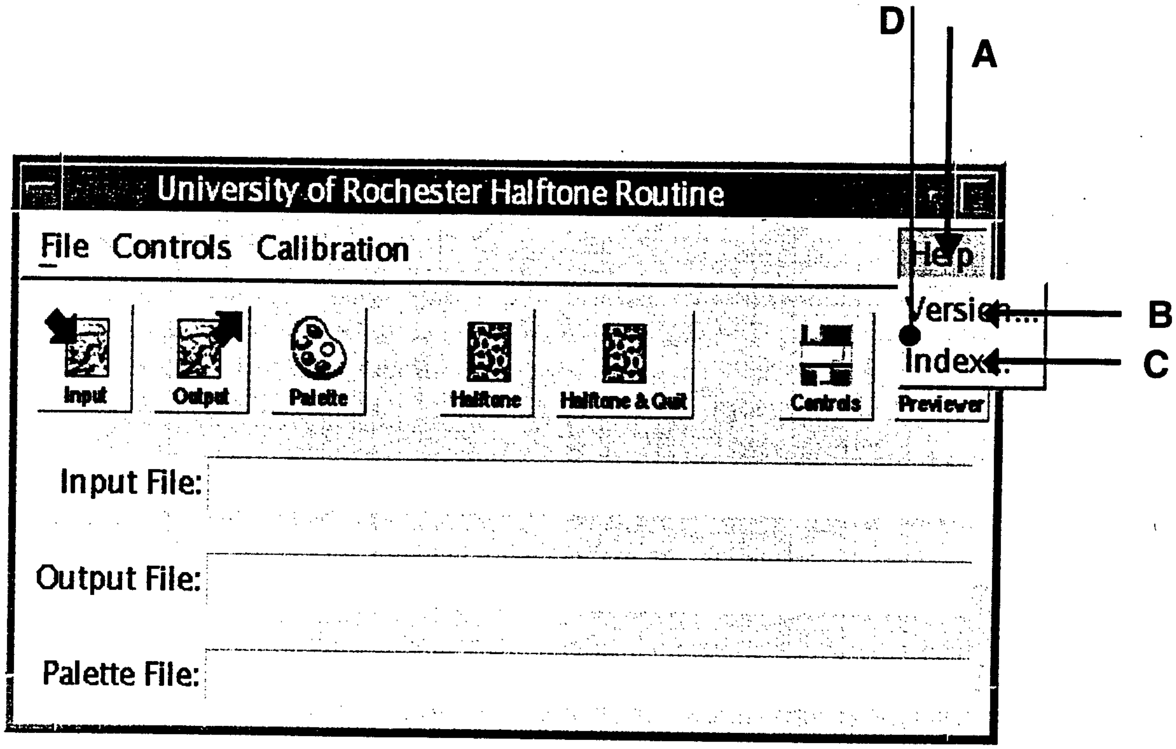
	Widget Names	Widget Classes
A	theControlsCascadeButton	xmCascadeButtonWidgetClass
B	theShowHideControlsMenuItem	xmPushButtonWidgetClass
C	theSaveSettingsMenuItem	xmPushButtonWidgetClass
D	theRecallSettingsMenuItem	xmPushButtonWidgetClass
E	theSetDefaultSettingMenuItem	xmPushButtonWidgetClass
F	theControlsPulldownMenu	xmRowColumnWidgetClass

4. The Calibration Menu


A
E
University of Rochester Halftone Routine
File
Controls
Calibration
Help
B
C
D
Generate Calibration Patches...
Make Calibration Table...
Select Calibration Table...
Input
Output
Controls
Previewer
Input File:
Output File:
Palette File:

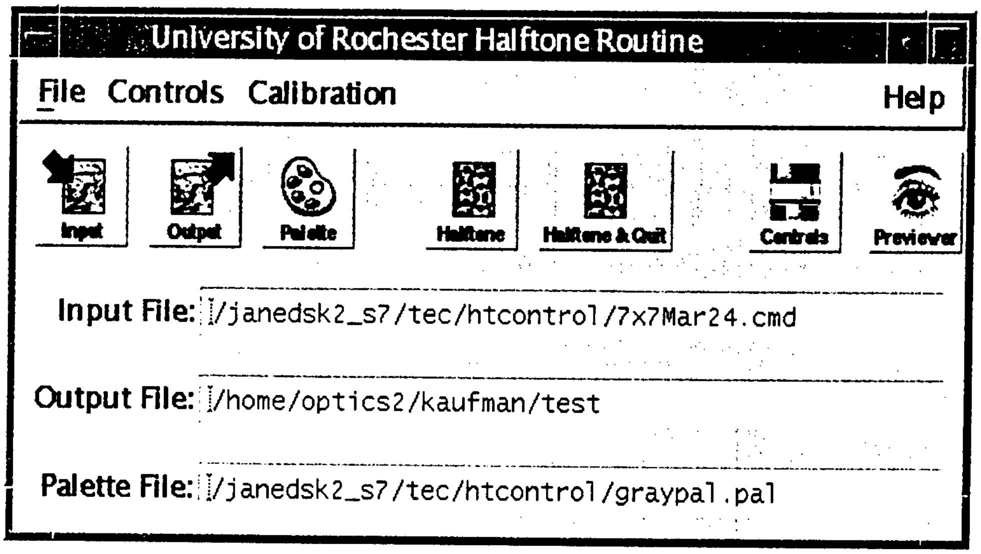
	Widget Names	Widget Classes
A	theCalibrationCascadeButton	xmCascadeButtonWidgetClass
B	theGeneratePatchesMenuItem	xmPushButtonWidgetClass
C	theMakeTableMenuItem	xmPushButtonWidgetClass
D	theRecallTableMenuItem	xmPushButtonWidgetClass
E	theCalibrationPulldownMenu	xmRowColumnWidgetClass

5. The Help Menu



	Widget Names	Widget Classes
A	theHelpCascadeButton	xmCascadeButtonWidgetClass
B	theVersionMenuItem	xmPushButtonWidgetClass
C	theIndexMenuItem	xmPushButtonWidgetClass
D	theHelpPulldownMenu	xmRowColumnWidgetClass

6. Palette Selection



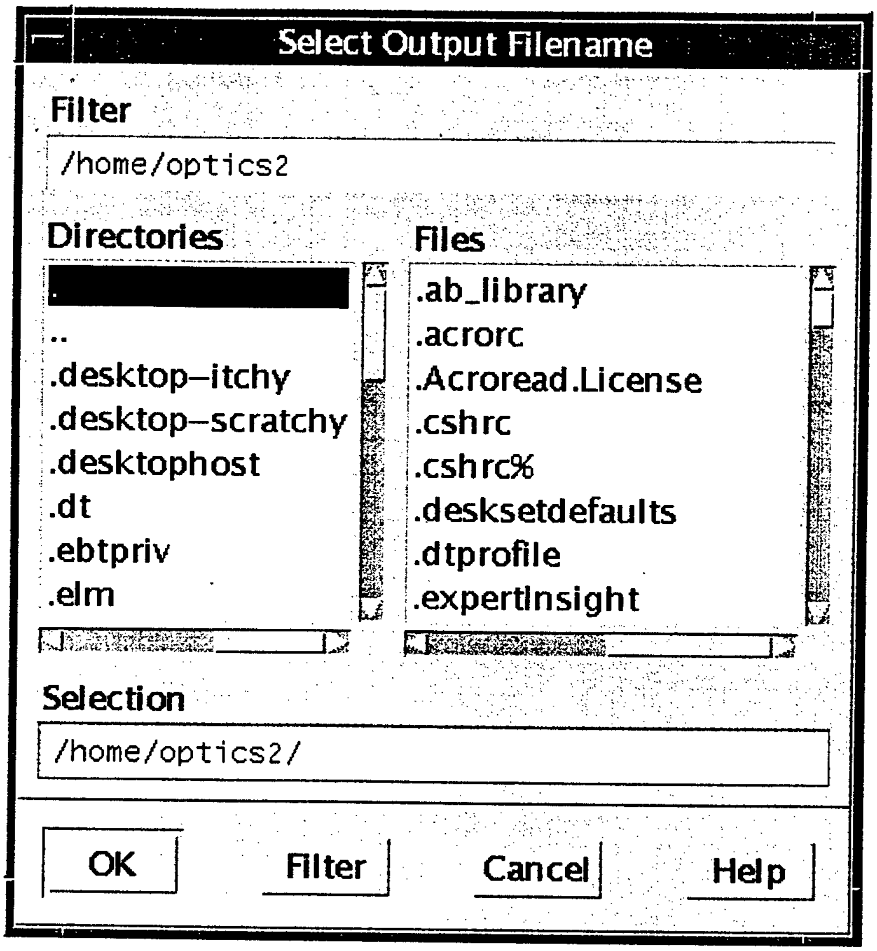
Input File: /janedsk2_s7/tec/htcontrol/7x7Mar24.cmd

Output File: /home/optics2/kaufman/test

Palette File: /janedsk2_s7/tec/htcontrol/graypal.pal

For certain file types the user can select a palette of colors. When a file of this type is entered into the Input File textfield the Palette pushbutton becomes selectable. The pushbutton will open a File-Selection Dialog Box so that the user can browse the choices of palette files. If the user already knows the palette file they can input it into the Palette File textfield

7. The File-Selection Dialog



Filter

/home/optics2

Directories

.
..
.desktop–itchy
.desktop–scratchy
.desktophost
.dt
.ebtpriv
.elm

Files

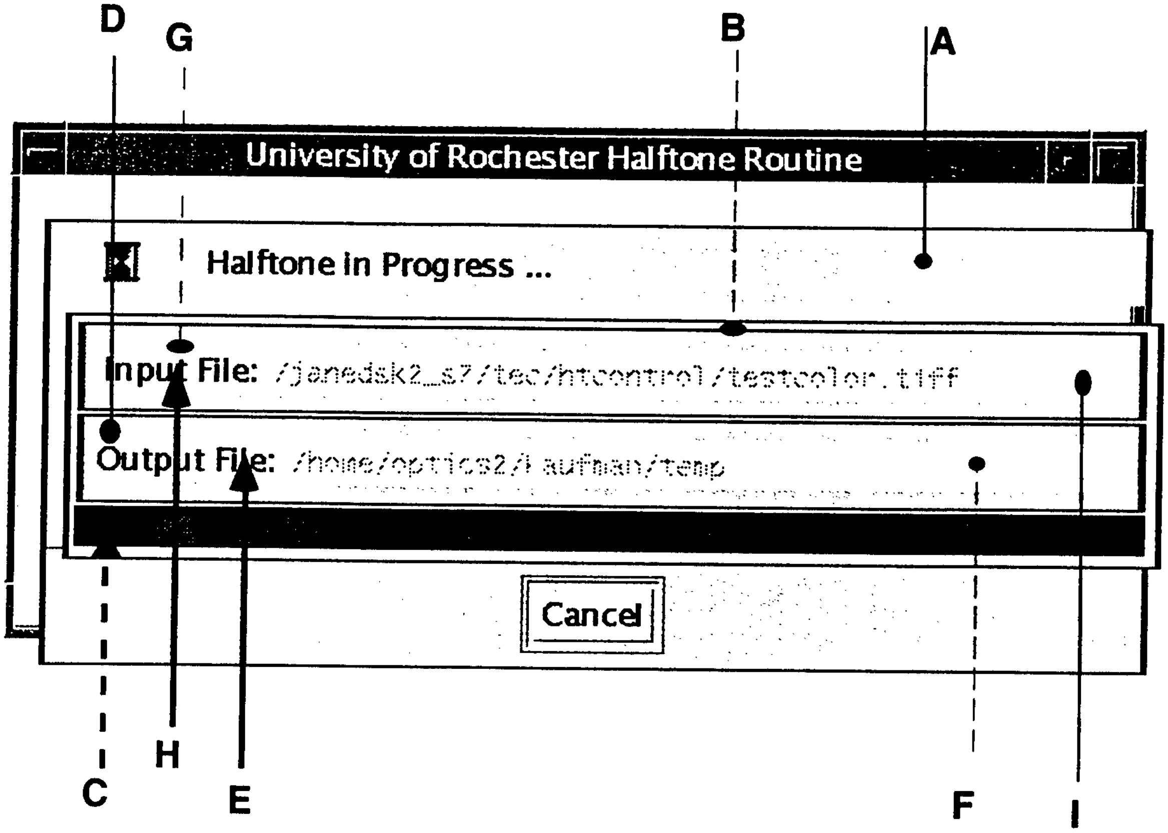
.ab_library
.acrorc
.Acroread.License
.cshrc
.cshrc%
.desksetdefaults
.dtprofile
.expertInsight

Selection

/home/optics2/

Motif provides a way of automating the task of selecting files with the File-Selection Dialog. The dialog allows the user to type the filename, or search through directories to find the file. Using the filter the user can specify what type of files they want to find. The File-Selection dialog can be altered to the users liking using the resource file. Htcontrol uses the standard File-Selection Dialog and resources which can be found in Ref. 1 and Ref. 2.

8. The Halftone Progress Window



University of Rochester Halftone Routine

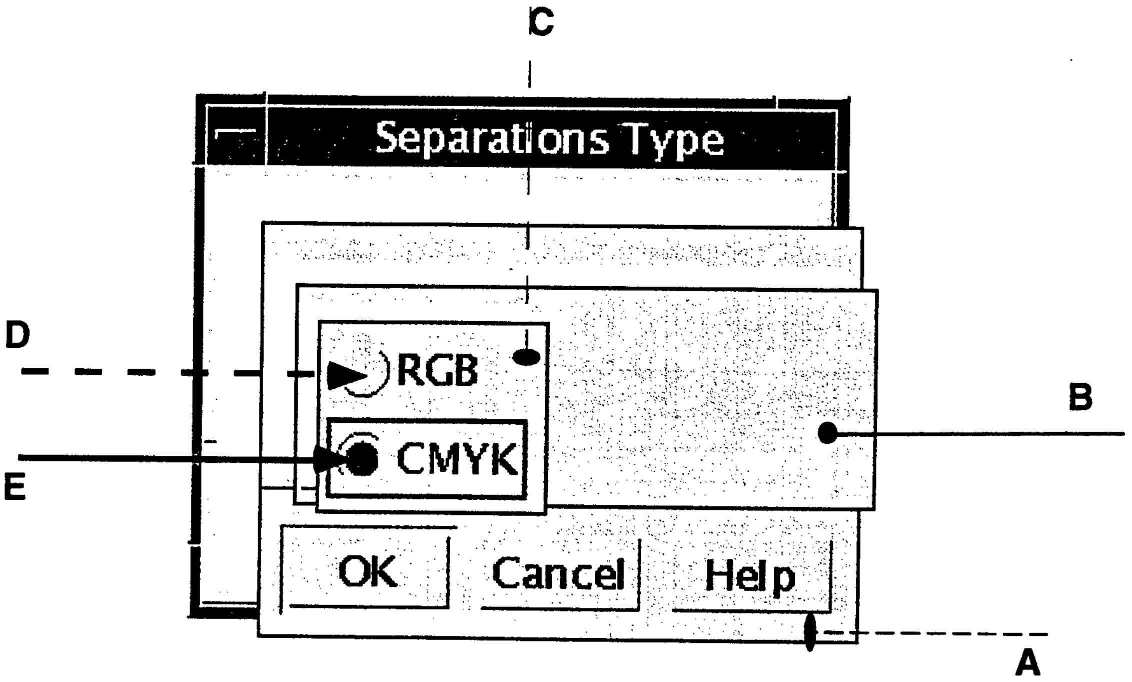
Halftone in Progress ...

Input File: /janedsk2_s7/tec/htcontrol/testcolor.tiff

Output File: /home/optics2/kaufman/temp

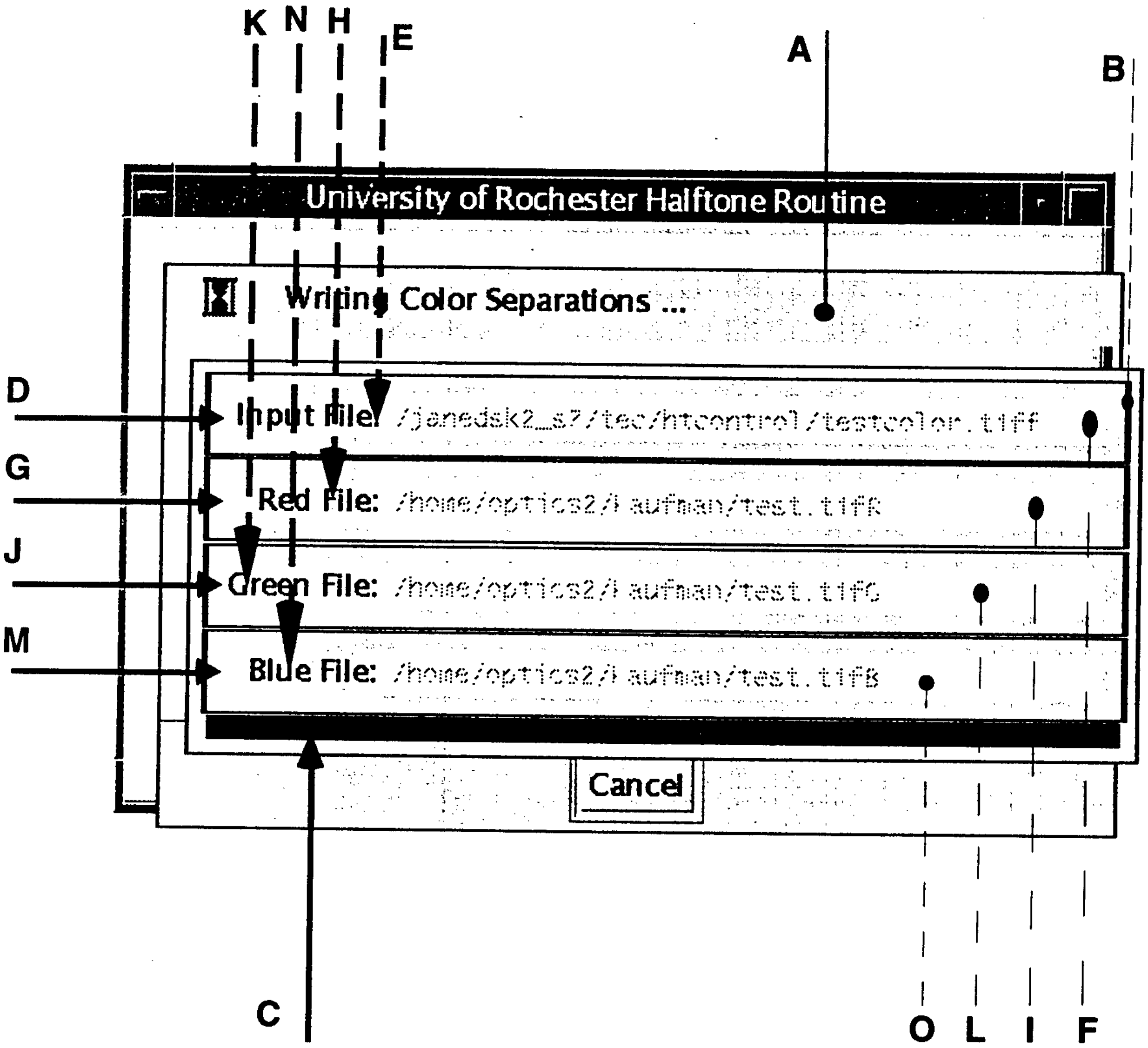
	Widget Names	Widget Classes
A	theHalftoneMessageBox	xmMessageBoxWidgetClass
B	theHalftoneInfoAreaForm	xmFormWidgetClass
C	theProgressScrollBar	xmScrolledBarWidgetClass
D	theHalftoneOutputFilenameForm	xmFormWidgetClass
E	theHalftoneOutputFilenameLabel	xmLabelWidgetClass
F	theHalftoneOutputFilenameTextField	xmTextFieldWidgetClass
G	theHalftoneInputFilenameForm	xmFormWidgetClass
H	theHalftoneInputFilenameLabel	xmLabelWidgetClass
I	theHalftoneInputFilenameTextField	xmTextFieldWidgetClass

9. The Save Separations Type Window



	Widget Names	Widget Classes
A	separationsTypeSelectionDialog	xmMessageBoxWidgetClass
B	theSeparationTypeSelectionForm	xmFormWidgetClass
C	theSeparationsTypeSelectionRadioBox	xmFormWidgetClass
D	theRGBSeparationsTypeToggleButton	xmToggleButtonWidgetClass
E	theCMYKSeparationsTypeToggleButton	xmToggleButtonWidgetClass

10. The RGB Color Separations Progress Window



University of Rochester Halftone Routine

Writing Color Separations ...

Input File: /janedsk2_s7/tec/htcontrol/testcolor.tiff

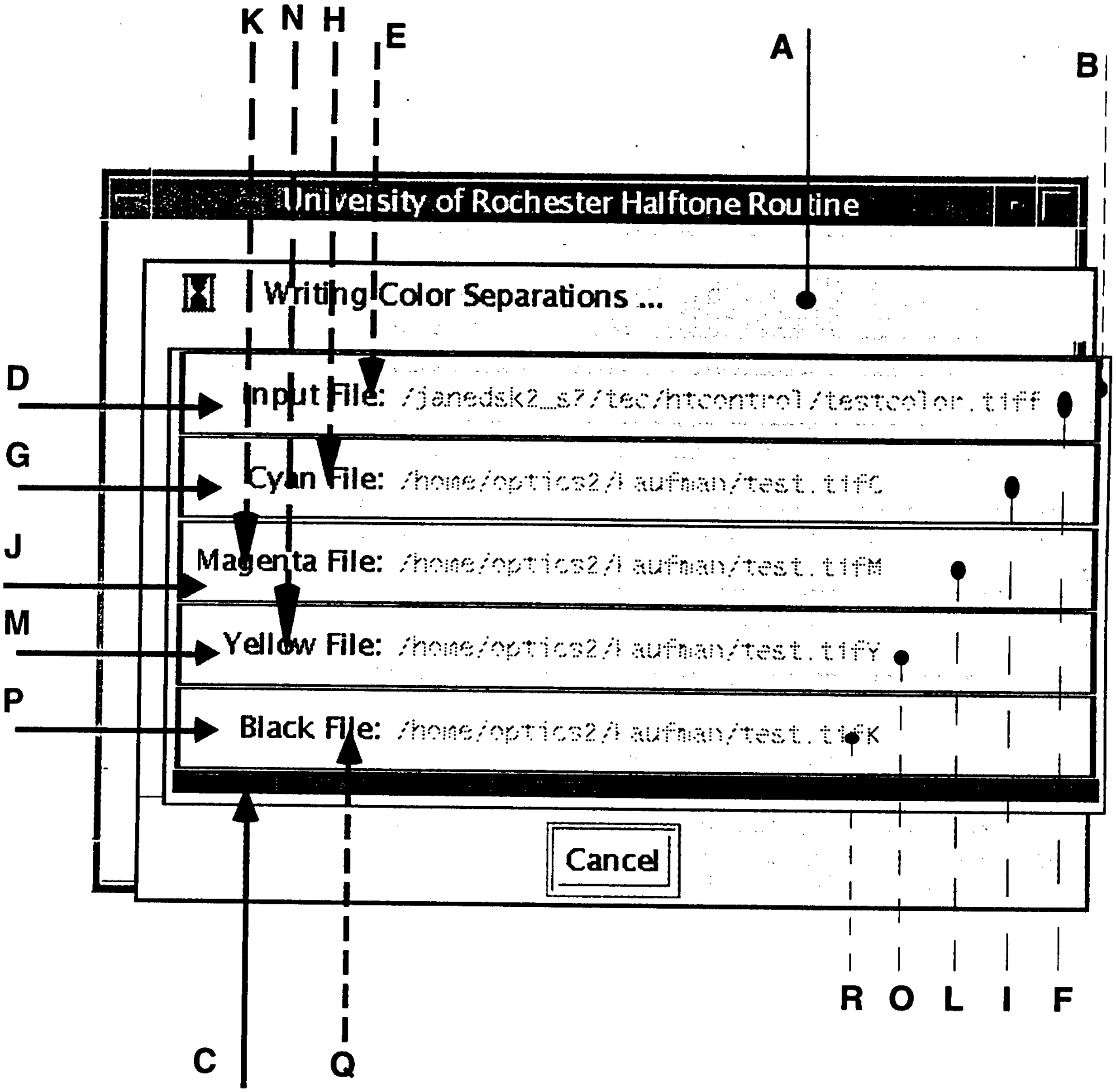
Red File: /home/optics2/kaufman/test.tiffR

Green File: /home/optics2/kaufman/test.tiffG

Blue File: /home/optics2/kaufman/test.tiffB

	Widget Names	Widget Classes
A	theSeparationsMessageBox	xmMessageBoxWidgetClass
B	theSeparationsInfoAreaForm	xmFormWidgetClass
C	theProgressScrollBar	xmScrolledBarWidgetClass
D	theSeparationsInputFilenameForm	xmFormWidgetClass
E	theSeparationsInputFilenameLabel	xmLabelWidgetClass
F	theSeparationsInputFilenameTextField	xmTextFieldWidgetClass
G	theRedOutputFilenameForm	xmFormWidgetClass
H	theRedOutputFilenameLabel	xmLabelWidgetClass
I	theRedOutputFilenameTextField	xmTextFieldWidgetClass
J	theGreenOutputFilenameForm	xmFormWidgetClass
K	theGreenOutputFilenameLabel	xmLabelWidgetClass
L	theGreenOutputFilenameTextField	xmTextFieldWidgetClass
M	theBlueOutputFilenameForm	xmFormWidgetClass
N	theBlueOutputFilenameLabel	xmLabelWidgetClass
O	theBlueOutputFilenameTextField	xmTextFieldWidgetClass

11. The CMYK Color Separations Progress Window



University of Rochester Halftone Routine

Writing Color Separations ...

Input File: /janedsk2_s7/tec/htcontrol/testcolor.tiff

Cyan File: /home/optics2/kaufman/test.tifC

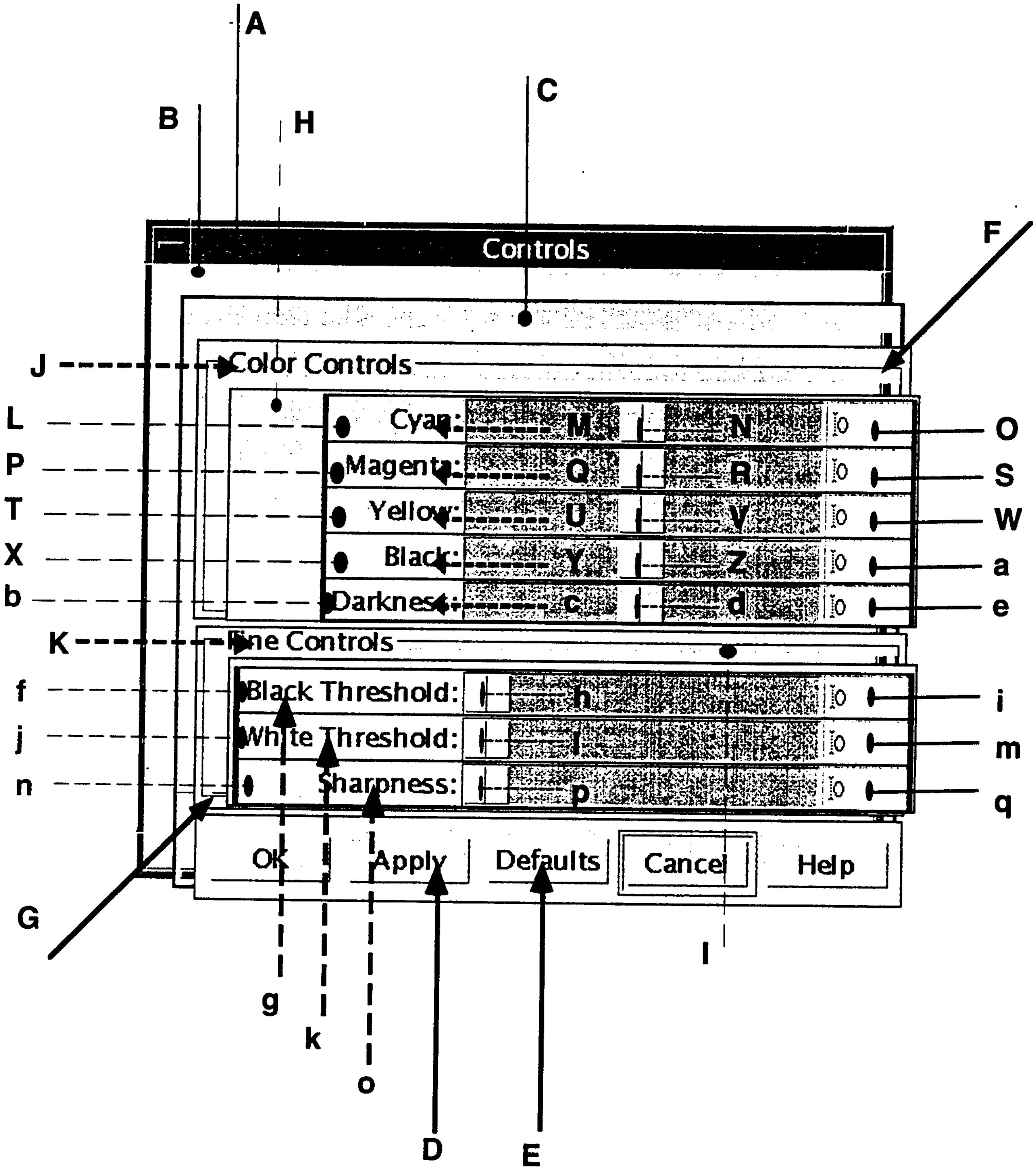
Magenta File: /home/optics2/kaufman/test.tifM

Yellow File: /home/optics2/kaufman/test.tifY

Black File: /home/optics2/kaufman/test.tifK

	Widget Names	Widget Classes
A	theSeparationsMessageBox	xmMessageBoxWidgetClass
B	theSeparationsInfoAreaForm	xmFormWidgetClass
C	theProgressScrollBar	xmScrolledBarWidgetClass
D	theSeparationsInputFilenameForm	xmFormWidgetClass
E	theSeparationsInputFilenameLabel	xmLabelWidgetClass
F	theSeparationsInputFilenameTextField	xmTextFieldWidgetClass
G	theCyanOutputFilenameForm	xmFormWidgetClass
H	theCyanOutputFilenameLabel	xmLabelWidgetClass
I	theCyanOutputFilenameTextField	xmTextFieldWidgetClass
J	theMagentaOutputFilenameForm	xmFormWidgetClass
K	theMagentaOutputFilenameLabel	xmLabelWidgetClass
L	theMagentaOutputFilenameTextField	xmTextFieldWidgetClass
M	theYellowOutputFilenameForm	xmFormWidgetClass
N	theYellowOutputFilenameLabel	xmLabelWidgetClass
O	theYellowOutputFilenameTextField	xmTextFieldWidgetClass
P	theBlackOutputFilenameForm	xmFormWidgetClass
Q	theBlackOutputFilenameLabel	xmLabelWidgetClass
R	theBlackOutputFilenameTextField	xmTextFieldWidgetClass

12. The Controls Window



Controls

Color Controls

Cyan: 0

Magenta: 0

Yellow: 0

Black: 0

Darkness: 0

Fine Controls

Black Threshold: 0

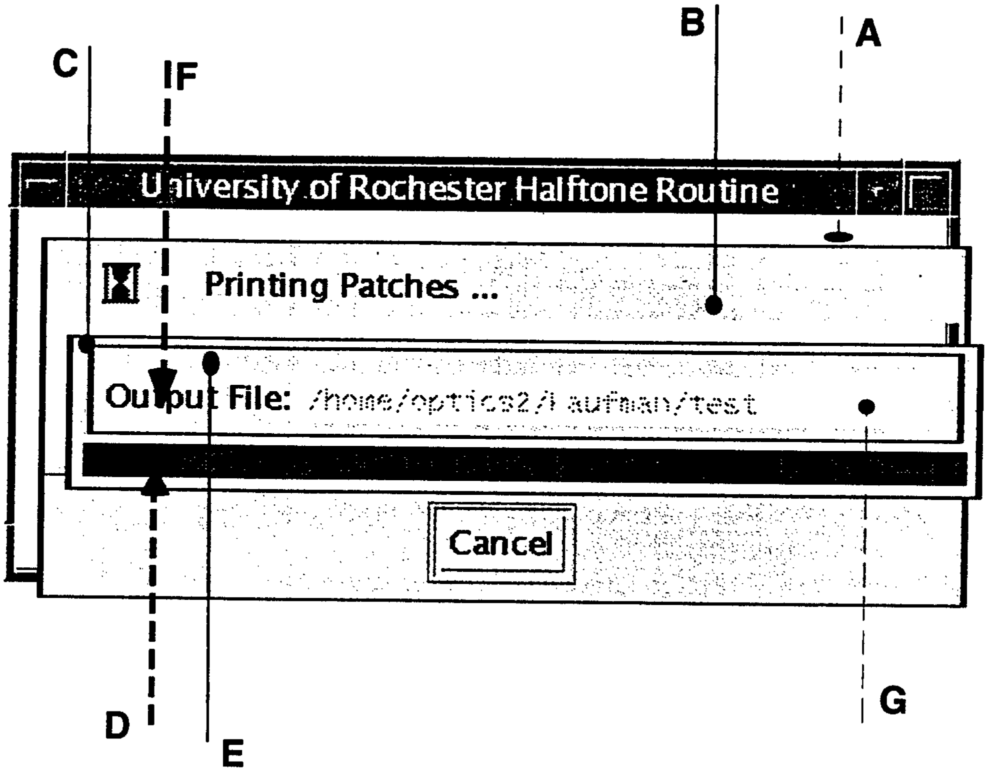
White Threshold: 0

Sharpness: 0

	Widget Names	Widget Classes
A	theControlsDialog	transientShellWidgetClass
B	theControlsMessageBox	xmMessageBoxWidgetClass
C	theControlsForm	xmFormWidgetClass
D	theControlsApplyButton	xmPushButtonWidgetClass
E	theControlsResetButton	xmPushButtonWidgetClass
F	theControlsColorFrame	xmFrameWidgetClass
G	theControlsFineFrame	xmFrameWidgetClass
H	theControlsColorForm	xmFormWidgetClass
I	theControlsFineForm	xmFormWidgetClass
J	theControlsColorLabel	xmLabelWidgetClass
K	theControlsFineLabel	xmLabelWidgetClass
L	theControlsCyanForm	xmFormWidgetClass
M	theControlsCyanLabel	xmLabelWidgetClass
N	theControlsCyanScale	xmScaleWidgetClass
O	theControlsCyanTextField	xmTextFieldWidgetClass
P	theControlsMagentaForm	xmFormWidgetClass
Q	theControlsMagentaLabel	xmLabelWidgetClass
R	theControlsMagentaScale	xmScaleWidgetClass
S	theControlsMagentaTextField	xmTextFieldWidgetClass
T	theControlsYellowForm	xmFormWidgetClass
U	theControlsYellowLabel	xmLabelWidgetClass

	Widget Names	Widget Classes
V	theControlsYellowScale	xmScaleWidgetClass
W	theControlsYellowTextField	xmTextFieldWidgetClass
X	theControlsBlackForm	xmFormWidgetClass
Y	theControlsBlackLabel	xmLabelWidgetClass
Z	theControlsBlackScale	xmScaleWidgetClass
a	theControlsBlackTextField	xmTextFieldWidgetClass
b	theControlsDarknessForm	xmFormWidgetClass
c	theControlsDarknessLabel	xmLabelWidgetClass
d	theControlsDarknessScale	xmScaleWidgetClass
e	theControlsDarknessTextField	xmTextFieldWidgetClass
f	theControlsBlackThresholdForm	xmFormWidgetClass
g	theControlsBlackThresholdLabel	xmLabelWidgetClass
h	theControlsBlackThresholdScale	xmScaleWidgetClass
i	theControlsBlackThresholdTextField	xmTextFieldWidgetClass
j	theControlsWhiteThresholdForm	xmFormWidgetClass
k	theControlsWhiteThresholdLabel	xmLabelWidgetClass
l	theControlsWhiteThresholdScale	xmScaleWidgetClass
m	theControlsWhiteThresholdTextField	xmTextFieldWidgetClass
n	theControlsSharpnessForm	xmFormWidgetClass
o	theControlsSharpnessLabel	xmLabelWidgetClass
p	theControlsSharpnessScale	xmScaleWidgetClass
q	theControlsSharpnessTextField	xmTextFieldWidgetClass

13. The Printing Patches Progress Window



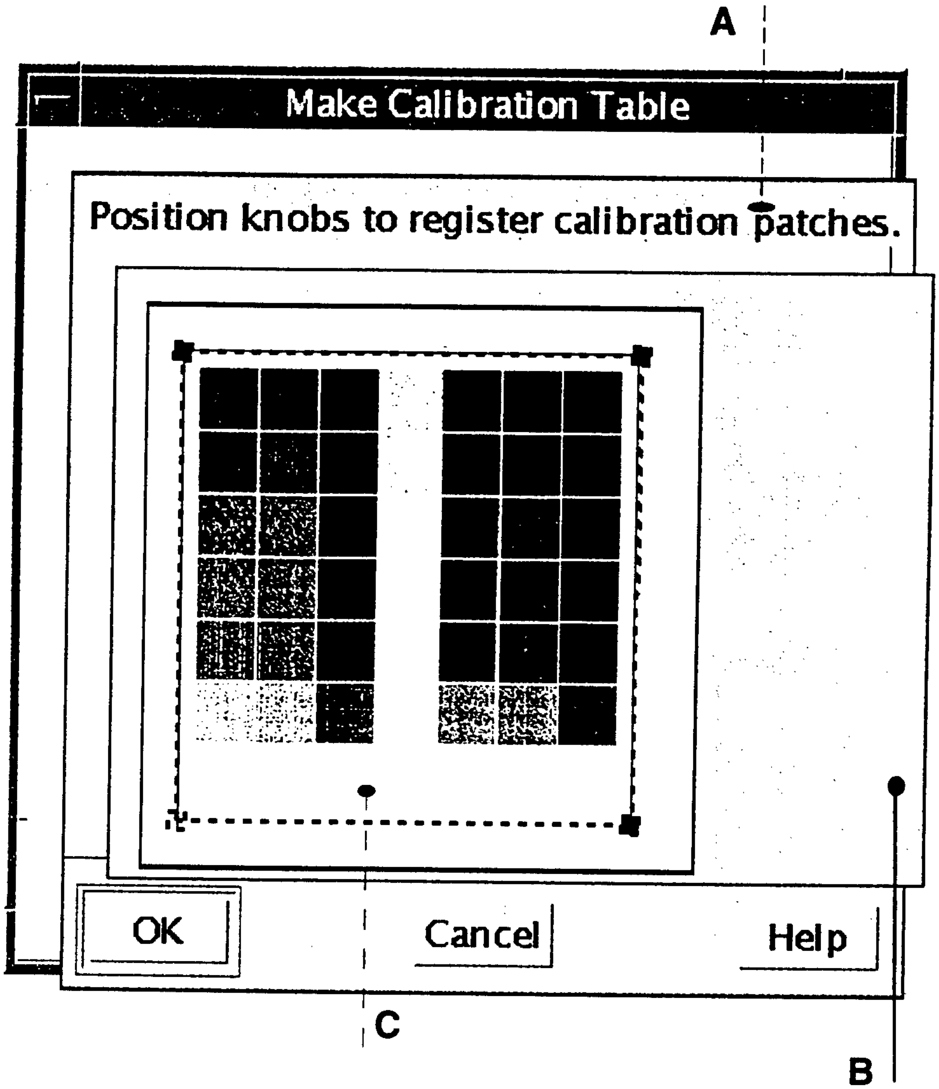
University of Rochester Halftone Routine

Printing Patches ...

Output File: /home/optics2/kaufman/test

	Widget Names	Widget Classes
A	theGeneratePatchesFileSelectionDialog	xmFileSelectionBoxWidgetClass
B	thePatchesMessageBox	xmMessageBoxWidgetClass
C	thePatchesInfoAreaForm	xmFormWidgetClass
D	theProgressScrollBar	xmScrolledBarWidgetClass
E	thePatchesOutputForm	xmFormWidgetClass
F	thePatchesOuputLabel	xmLabelWidgetClass
G	thePatchesOuputTextField	xmTextFieldWidgetClass

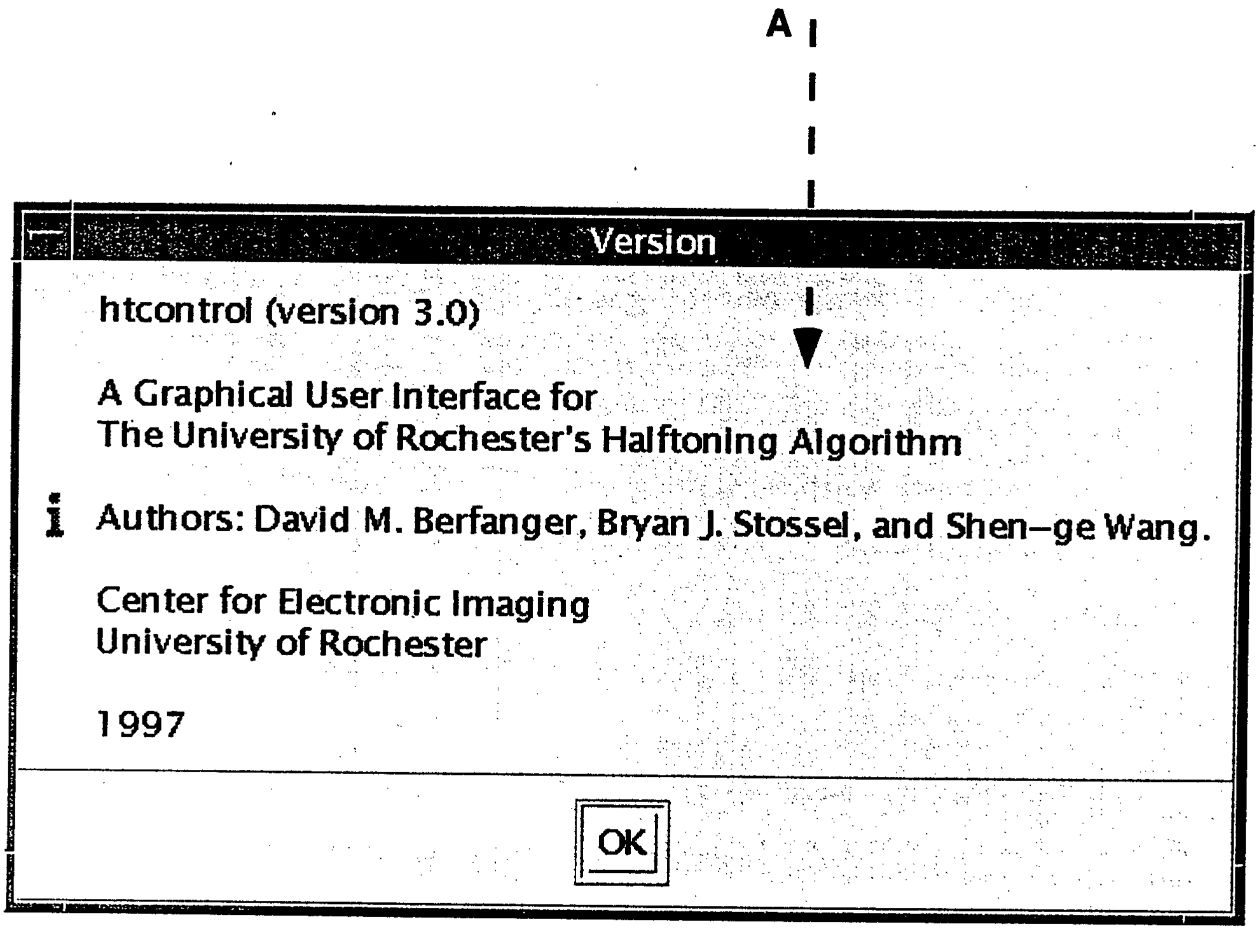
14. The Make Callibration Table Window



Position knobs to register calibration patches.

	Widget Names	Widget Classes
A	theCalibrationDialogMessageBox	xmMessageBoxWidgetClass
B	aCalibrationDialogForm	xmFormWidgetClass
C	aCalibrationImageView	vImageViewWidgetClass

15. The Version Menu Window



htcontrol (version 3.0)

A Graphical User Interface for
The University of Rochester's Halftoning Algorithm

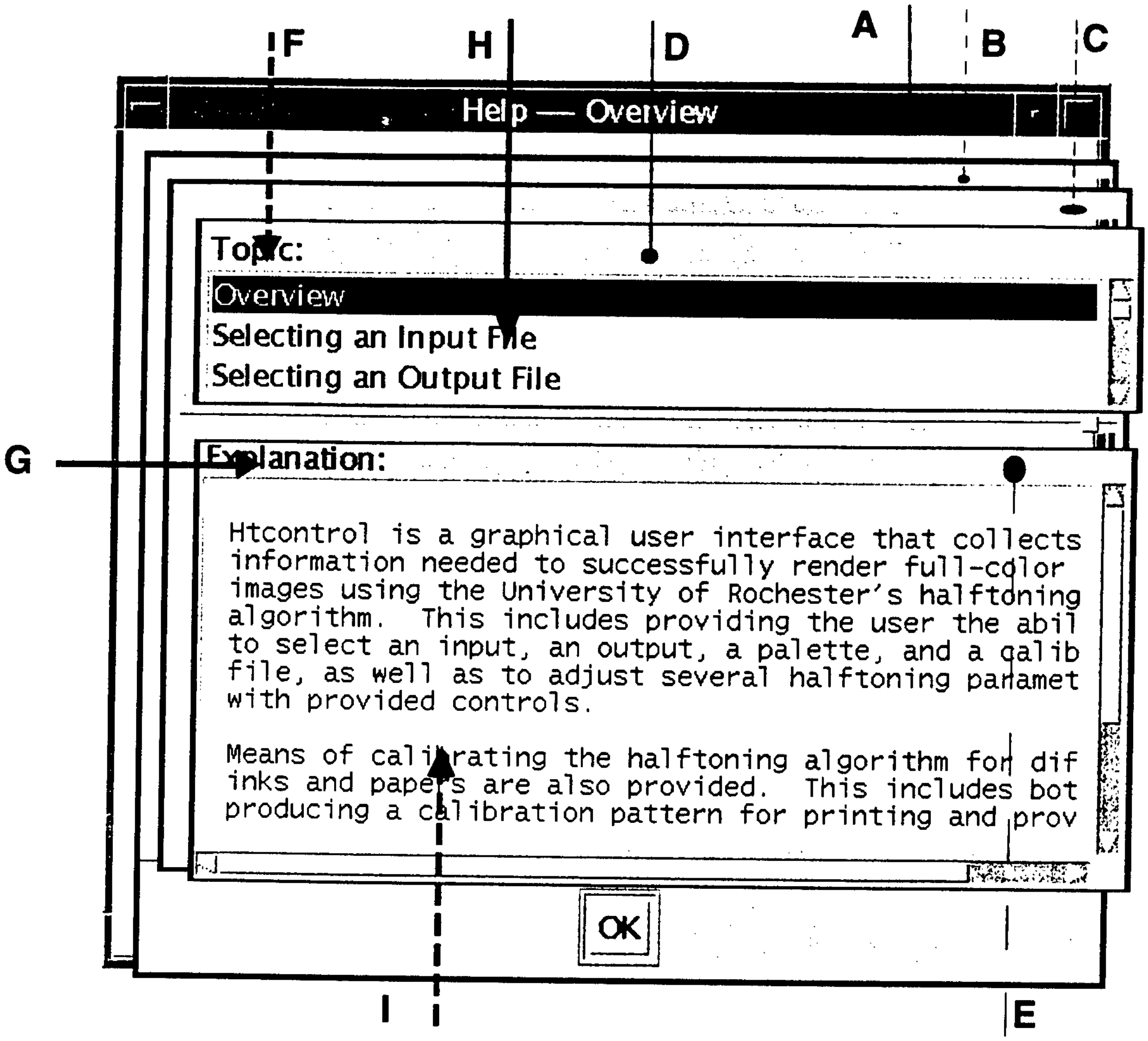
Authors: David M. Berfanger, Bryan J. Stossel, and Shen–ge Wang.

Center for Electronic Imaging
University of Rochester

1997

	Widget Names	Widget Classes
A	theVersionDialog	xmMessageBoxWidgetClass

16. The Help Dialog Window.

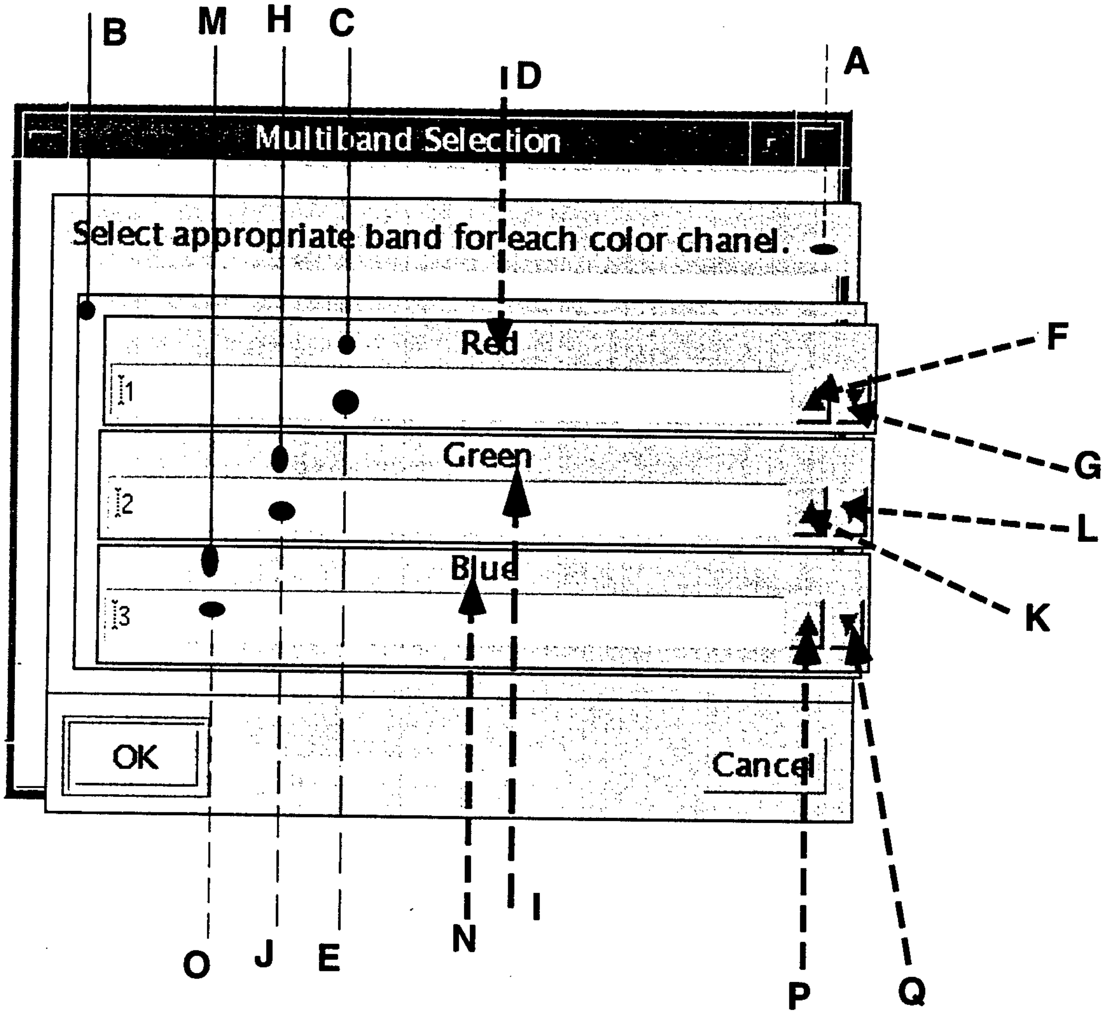


```
Htcontrol is a graphical user interface that collects
information needed to successfully render full-color
images using the University of Rochester's halftoning
algorithm.  This includes providing the user the abil
to select an input, an output, a palette, and a calib
file, as well as to adjust several halftoning paramet
with provided controls.

Means of calibrating the halftoning algorithm for dif
inks and papers are also provided.  This includes bot
producing a calibration pattern for printing and prov
```

	Widget Names	Widget Classes
A	theHelpMessageBox	xmMessageBoxWidgetClass
B	theHelpDialog	topLevelShellWidgetClass
C	theHelpDialogPanedWindow	xmPanedWindowWidgetClass
D	theHelpDialogTopicsForm	xmFormWidgetClass
E	theHelpDialogTextForm	xmFormWidgetClass
F	theHelpDialogTopicsLabel	xmLabelWidgetClass
G	theHelpDialogTextLabel	xmLabelWidgetClass
H	theHelpDialogTopicsList	xmListWidgetClass
I	theHelpDialogScrolledText	xmTextedWidgetClass

17. The Mutliband Selection Box Window



Multiband Selection

Select appropriate band for each color chanel.

Red

1

Green

2

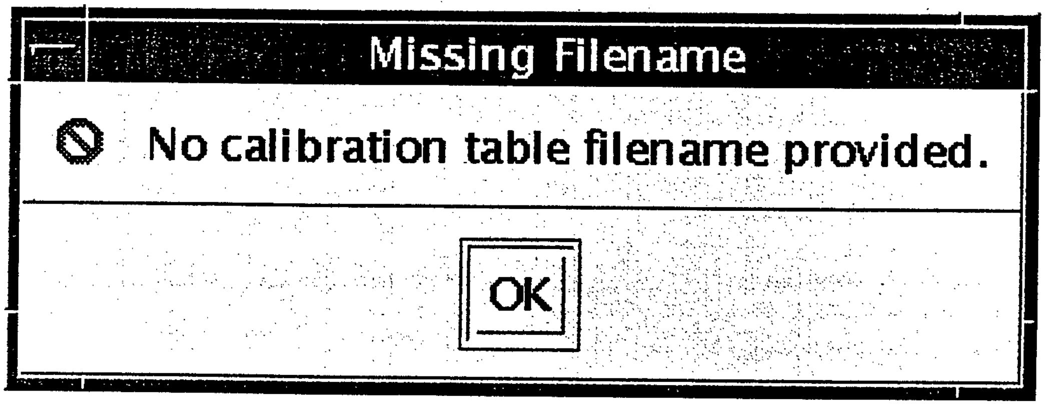
Blue

3

	Widget Names	Widget Classes
A	theMutlibandSelectionDialog	xmMessageBoxWidgetClass
B	theMultibandSelectionDialog_geometryManagerForm	xmFormWidgetClass
C	theRedSpinBoxForm	xmFormWidgetClass
D	theRedSpinBoxLabel	xmLabelWidgetClass
E	theRedSpinBoxTextField	xmTextFieldWidgetClass
F	theRedIncrementArrowButton	xmPushButtonWidgetClass
G	theRedDecrementArrowButton	xmPushButtonWidgetClass
H	theGreenSpinBoxForm	xmFormWidgetClass
I	theGreenSpinBoxLabel	xmLabelWidgetClass
J	theGreenSpinBoxTextField	xmTextFieldWidgetClass
K	theGreenIncrementArrowButton	xmPushButtonWidgetClass
L	theGreenDecrementArrowButton	xmPushButtonWidgetClass
M	theBlueSpinBoxForm	xmFormWidgetClass
N	theBlueSpinBoxLabel	xmLabelWidgetClass
O	theBlueSpinBoxTextField	xmTextFieldWidgetClass
P	theBlueIncrementArrowButton	xmPushButtonWidgetClass
Q	theBlueDecrementArrowButton	xmPushButtonWidgetClass

18. Standard Error Message

Error Dialog

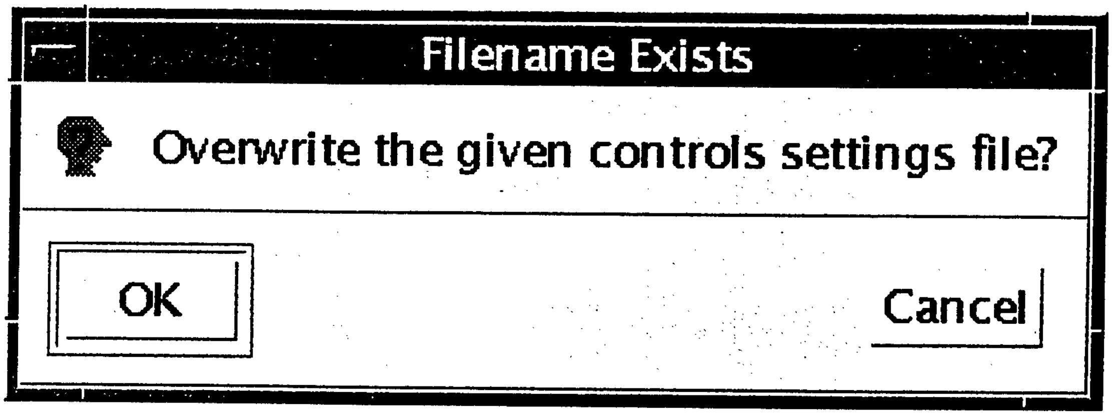


No calibration table filename provided.

An error dialog window is created using a standard dialog. A standard dialog consists of three parts: the text message, an icon showing what kind of dialog is being displayed and three pushbuttons(OK, Cancel, and Help). In the error dialog above there is only the OK button so the Cancel and Help button are removed. This dialog displays an error message and disappears when the user clicks on OK.

19. Standard Question Message

Question Dialog



Filename Exists

Overwrite the given controls settings file?

A question dialog window is also created using a standard dialog. A standard dialog consists of three parts: the text message, an icon showing what kind of dialog is being displayed and three pushbuttons(OK, Cancel, and Help). In the question dialog above the Help button is removed. This dialog displays a question which the user can agree with by pressing the OK buttton or disagree with by pressing the Cancel button.

APPENDIX C: Error Messages

readCalibrationPatchesError

calibrationPatchesFileFormatError

separationsFilenameExists

noInputFilename

noOutputFilename

outputFilenameExists

invalidPaletteFilename

unwritableCalibrationTableFilename

unreadableCalibrationTableFilename

unreadableCalibrationPatchFilename

unwritableCalibrationPatchFilename

unwritableControlsSettingsFilename

unreadableControlsSettingsFilename

unwritableSeparationsFilename

invalidHalftoneExecutableFilename

noSeparationsExecutableFilename

invalidSeparationsExecutableFilename

noPatchesExecutableFilename

invalidPatchesExecutableFilename

writeDefaultControlsSettingsError

calibrationPatchFileFormatError

tiffOpenError

unkownInputFileTypeError

theHalftoneInputFilenameLabel

theHalftoneOutputFilenameLabel

maketableError

writeError

unwritableSeparationsFilename

invalidInputFilename

invalidOutputFilename

noPaletteFilename

noCalibrationTableFilename

noCalibrationPatchFilename

calibrationTableFilenameExists

noControlsSettingsFilename

calibrationPatchFilenameExists

controlsSettingsFilenameExists

noSeparationsFilename

noHalftoneExecutableFilename

writeControlsSettingsError

readCalibrationPatchError

outputOpenError

inputOpenError

erdasOpenError

paletteReadError

tangentOpenError

tiffInitError

separationsWriteError

theRedSpinBoxLabel

theGreenSpinBoxLabel

theHalftoneInputFilenameTextField

theHalftoneOutputFilenameTextField

theMultibandSelectionDialog

outputWriteError

tangentHalftoningError

theBlueSpinBoxLabel

overlapReadError

calibrationReadError

erdasHalftoningError

tiffHalftoningError

APPENDIX D: Application Resources

XtNdefaultCyanValue

XtCDefaultCyanValue

XtNdefaultMagentaValue

XtCDefaultMagentaValue

XtNdefaultYellowValue

XtCDefaultYellowValue

XtNdefaultBlackValue

XtCDefaultBlackValue

XtNdefaultDarknessValue

XtCDefaultDarknessValue

XtNdefaultBlackThresholdValue

XtCDefaultBlackThresholdValue

XtNdefaultWhiteThresholdValue

XtCDefaultWhiteThresholdValue

XtNdefaultSharpnessValue

XtCDefaultSharpnessValue

XtNcyanValue

XtCCyanValue

XtNmagentaValue

XtCMagentaValue

XtNyellowValue

XtCYellowValue

XtNblackValue

XtCBlackValue

XtNdarknessValue

XtCDarknessValue

XtNblackThresholdValue

XtCBlackThresholdValue

XtNwhiteThresholdValue

XtCWhiteThresholdValue

XtNsharpnessValue

XtCSharpnessValue

XtNcalibrationFilename

XtCCalibrationFilename

XtNpaletteFilename

XtCPaletteFilename

XtNdefaultSettingsFilename

XtCDefaultSettingsFilename

XtNhalftoneExecFilename

XtCHalftoneExecFilename

XtNseparationsExecFilename

XtCSeparationsExecFilename

XtNpatchesExecFilename

XtCPatchesExecFilename

XtNhelpFilename

XtCHelpFilename

XtNtangentFilenameExtension

XtCTangentFilenameExtension

XtNtiffFilenameExtension

XtCTiffFilenameExtension

XtNerdasFilenameExtension

XtCErdasFilenameExtension

XtNredSeparationFilenameExtension

XtCRedSeparationFilenameExtension

XtNredSeparationFilenameExtension

XtCRedSeparationFilenameExtension

XtNgreenSeparationFilenameExtension

XtCGreenSeparationFilenameExtension

XtNblueSeparationFilenameExtension

XtCBlueSeparationFilenameExtension

XtNcyanSeparationFilenameExtension

XtCCyanSeparationFilenameExtension

XtNmagentaSeparationFilenameExtension

XtCMagentaSeparationFilenameExtension

XtNyellowSeparationFilenameExtension

XtCYellowSeparationFilenameExtension

XtNblackSeparationFilenameExtension

XtCBlackSeparationFilenameExtension

XtNcalibrationFilenameExtension

XtCCalibrationFilenameExtension

XtNpaletteFilenameExtension

XtCPaletteFilenameExtension

XtNseparationsType

XtCSeparationsType

APPENDIX E: Example Resource File - Htcontrol 3.0